Institut Teknologi Sepuluh Nopember

Departemen Sistem Informasi
Mata Kuliah : Rekayasa Kebutuhan Perangkat Lunak

Use Case

Rarasmaya Indraswari – raras@its.ac.id

Use-Case

- **Use case** is a list of actions or event steps typically defining the interactions between a role (known in the Unified Modeling Language (UML) as an *actor*) and a system / process to achieve a goal
- Use case diagrams are used to visualize, specify, construct, and document the (intended) behavior of the system, during requirements capture and analysis.
- Provide a way for developers, domain experts and end-users to communicate.
- Serve as basis for testing.
- Use case diagrams contain use cases, actors, and their relationships.

Use-Case Actor

- An actor is a system or role interacting with the system under development.
- You should include a **system as an actor** in a use case if it is outside the system you are developing, and if it directly interacts with the system you are developing.
- This is important because you need to define the boundary of your system, which means its scope and interfaces
- Including a system as an actor will clearly state the requirement for your system under development to provide a suitable interface for that actor system.

Use-Case Relationship

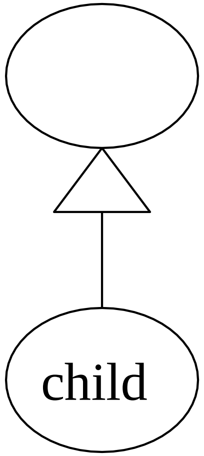
1) Generalization - use cases that are specialized versions of other use cases.

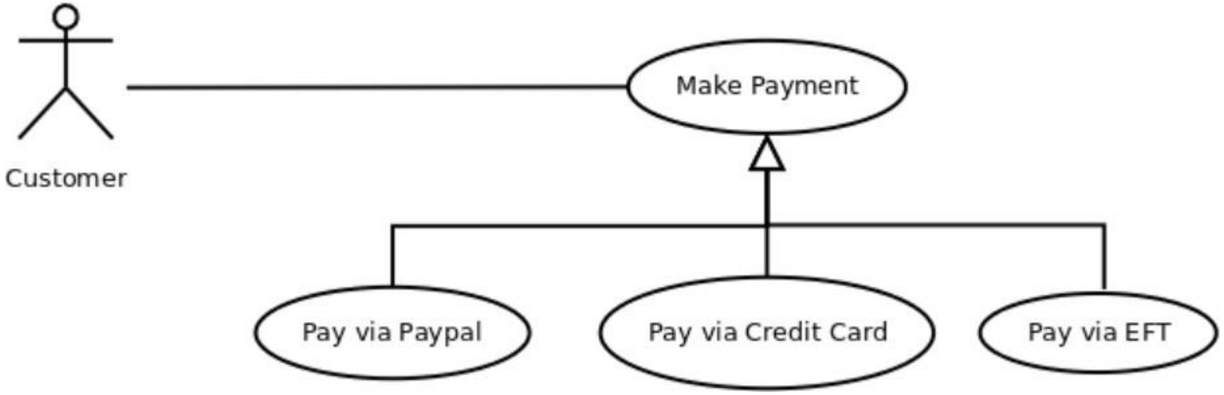
2) Include - use cases that are included as parts of other use cases. Enable to factor common behavior.

3) Extend - use cases that extend the behavior of other core use cases. Enable to factor variants.

1) Generalization

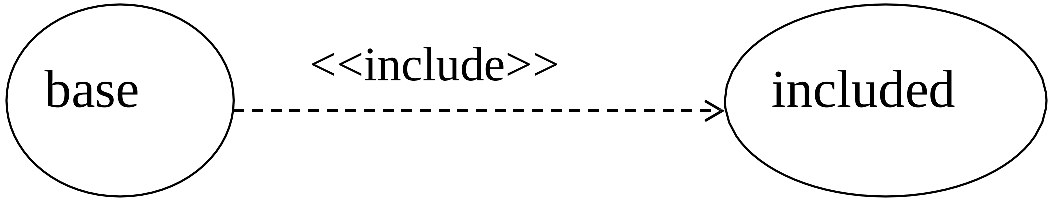
- The child use case inherits the behavior and meaning of the parent use case.
- The child may add to or override the behavior of its parent.



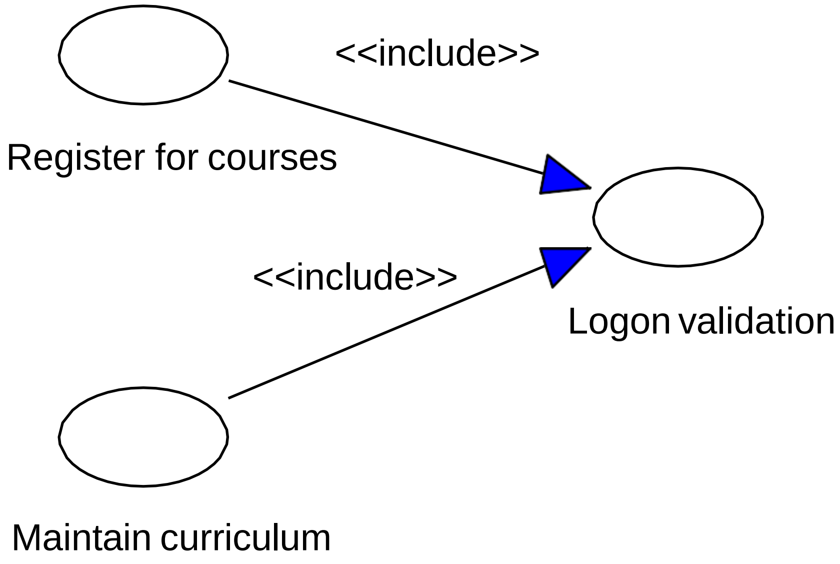


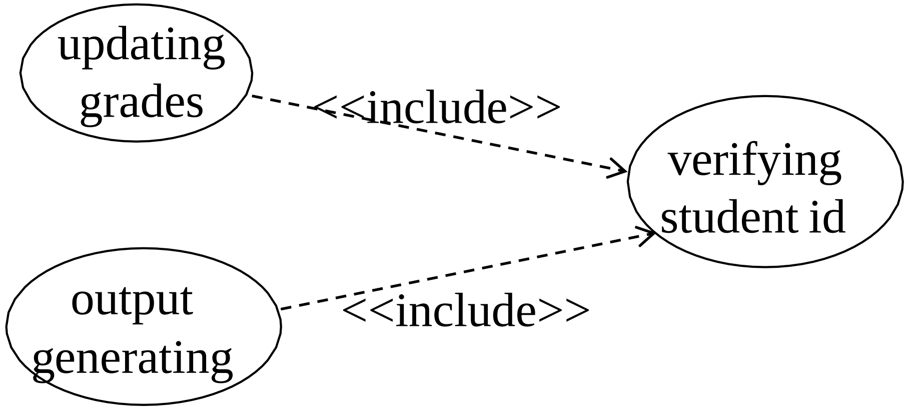
2) Include

The included use case never stands alone. It only occurs as a part of some larger base that includes it.



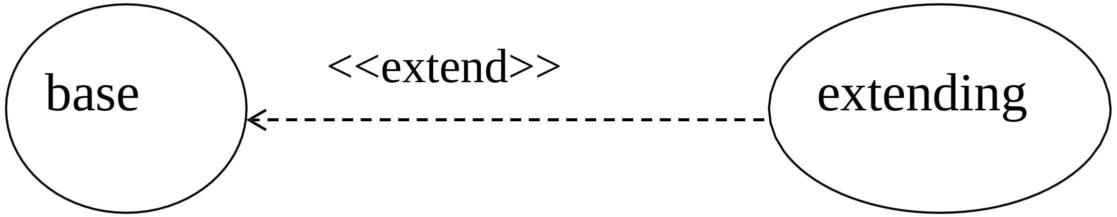
Example of Include

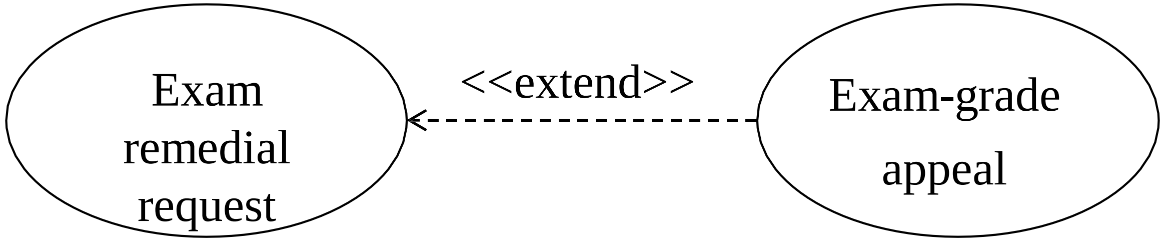




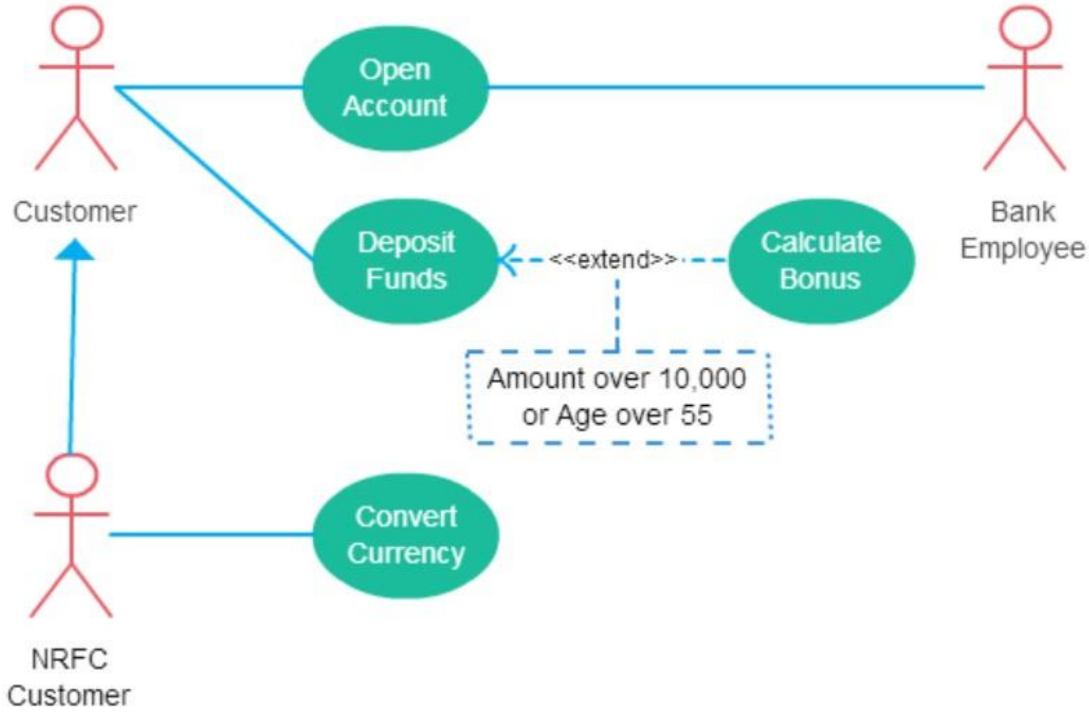
3) Extend

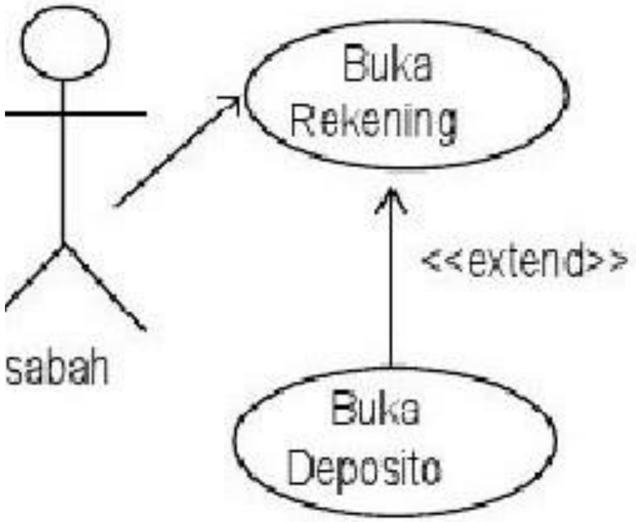
- The base use case may stand alone, but under certain conditions its behavior may be extended by the behavior of another use case.
- Enables to model optional behavior or branching under conditions





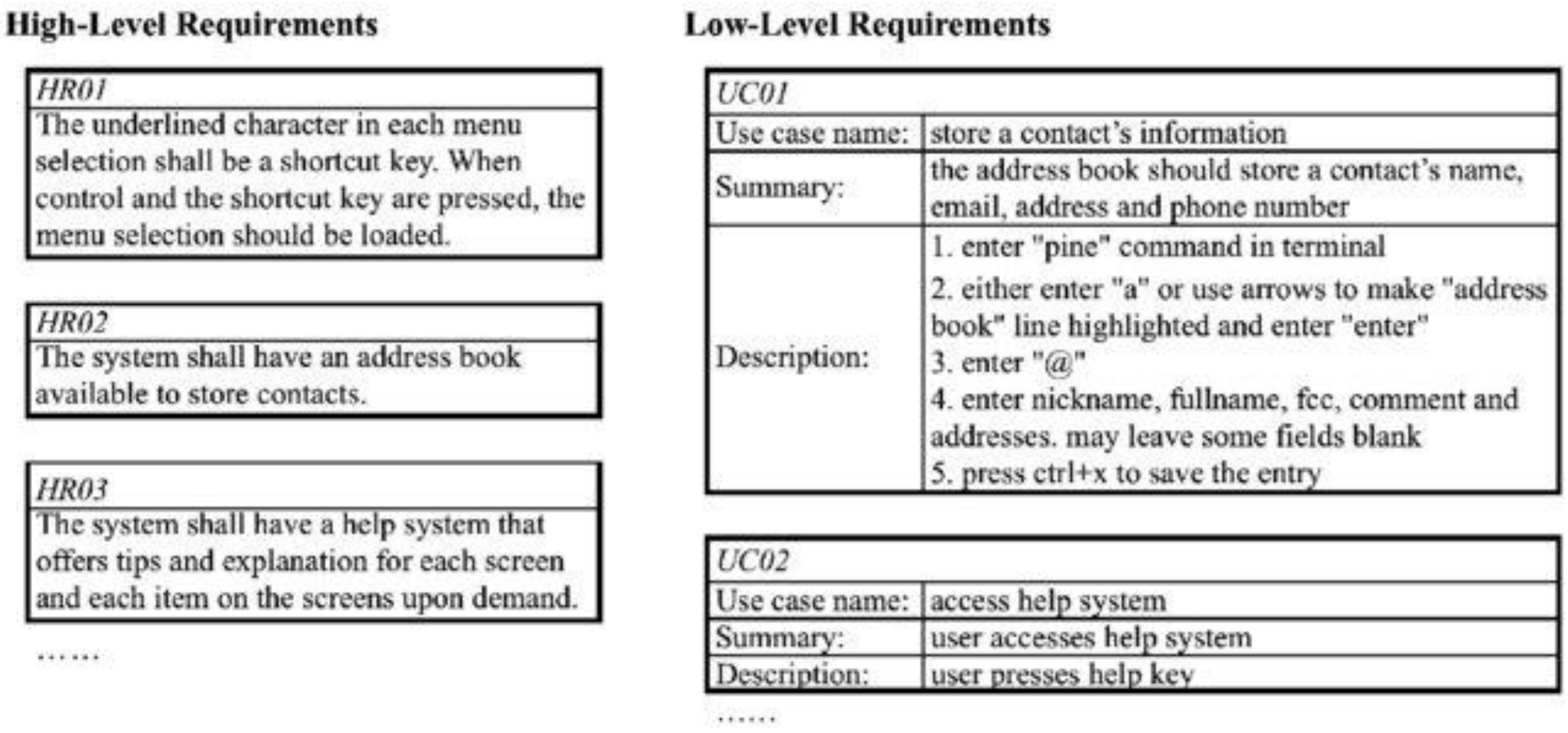
Example of Extend





Use-Case & Functional Requirements

- Use cases **are** the functional requirements
- Use cases **reveal** the functional requirements



High-Level Requirements

HR01
The underlined character in each menu selection shall be a shortcut key. When control and the shortcut key are pressed, the menu selection should be loaded.

HR02
The system shall have an address book available to store contacts.

HR03
The system shall have a help system that offers tips and explanation for each screen and each item on the screens upon demand.

……

Low-Level Requirements

UC01	
Use case name:	store a contact's information
Summary:	the address book should store a contact's name, email, address and phone number
Description:	1. enter "pine" command in terminal 2. either enter "a" or use arrows to make "address book" line highlighted and enter "enter" 3. enter "@" 4. enter nickname, fullname, fcc, comment and addresses. may leave some fields blank 5. press ctrl+x to save the entry

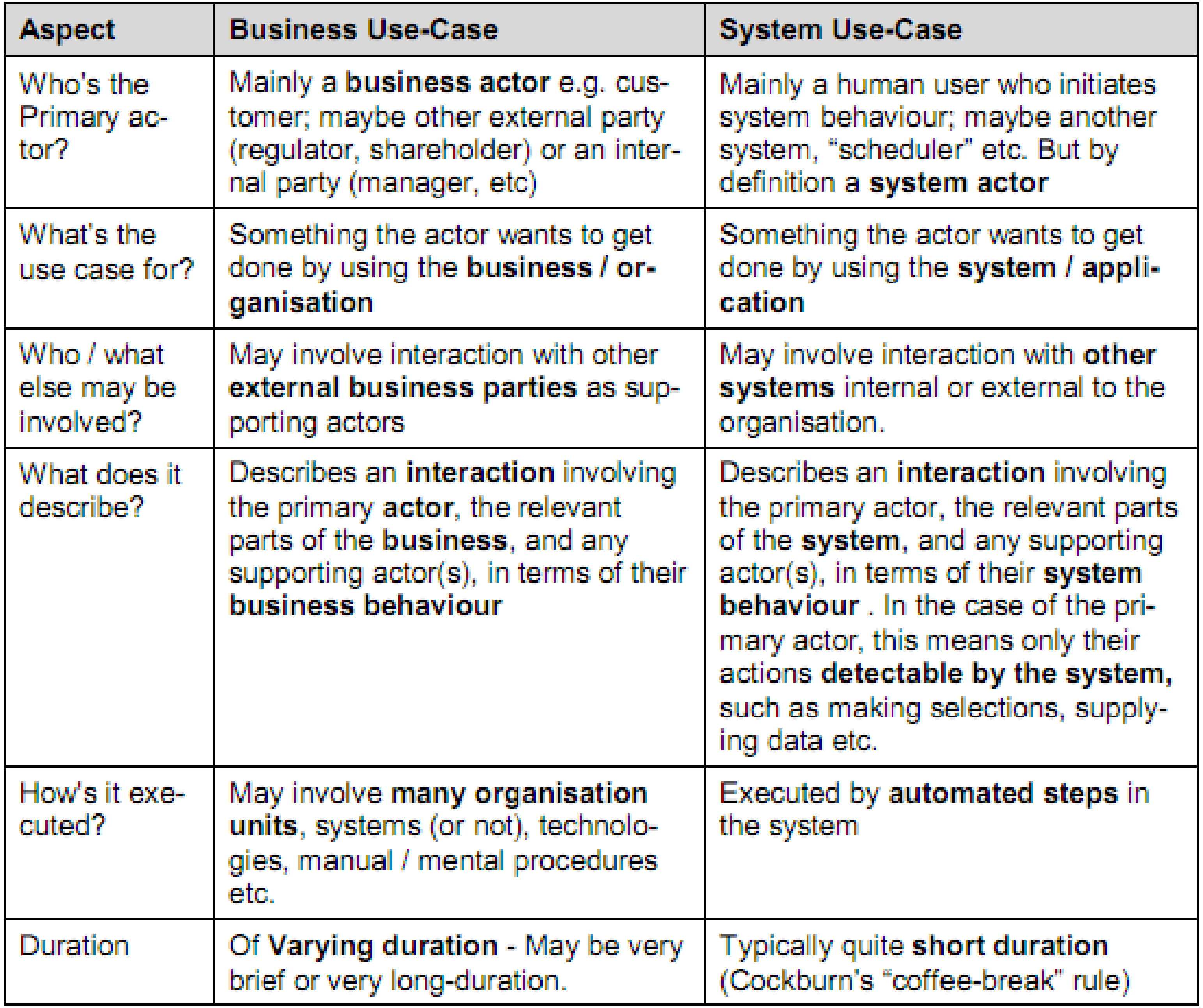
UC02	
Use case name:	access help system
Summary:	user accesses help system
Description:	user presses help key

……

Use Case

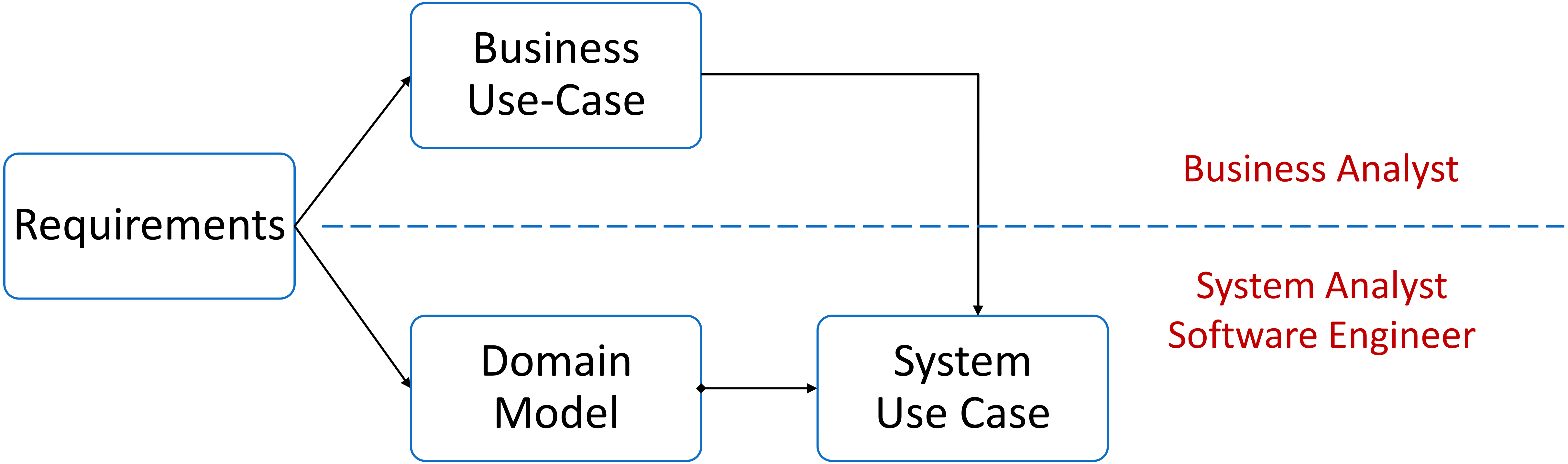
1. **Business Use-Case**: describes the steps in a process that achieve a business goal
2. **System Use-Case / Use-Case Diagram**: describes behavior of a system that automates a business use-case

Use-Case



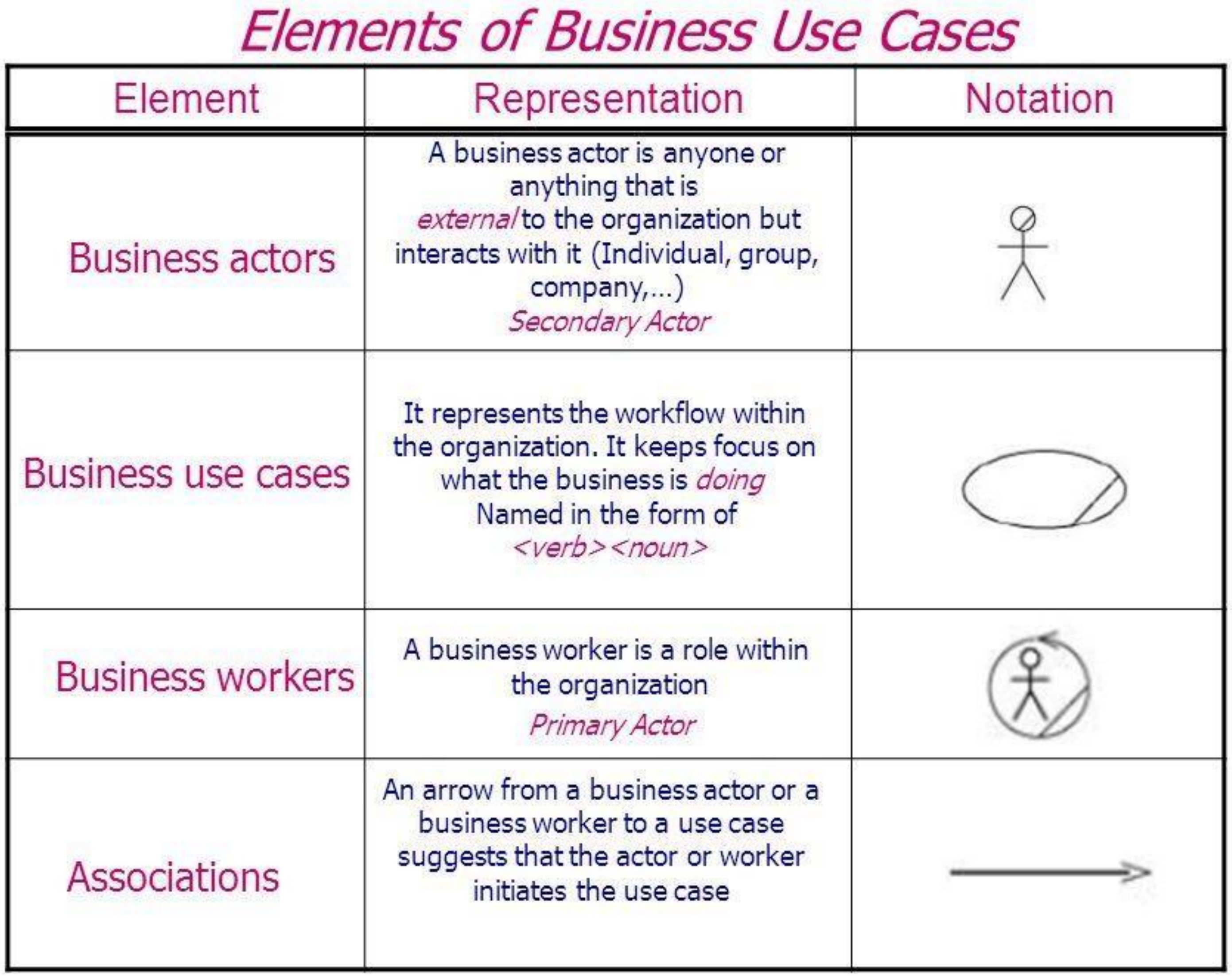
Aspect	Business Use-Case	System Use-Case
Who's the Primary actor?	Mainly a **business actor** e.g. customer; maybe other external party (regulator, shareholder) or an internal party (manager, etc)	Mainly a human user who initiates system behaviour; maybe another system, "scheduler" etc. But by definition a **system actor**
What's the use case for?	Something the actor wants to get done by using the **business / organisation**	Something the actor wants to get done by using the **system / application**
Who / what else may be involved?	May involve interaction with other **external business parties** as supporting actors	May involve interaction with **other systems** internal or external to the organisation.
What does it describe?	Describes an **interaction** involving the primary **actor**, the relevant parts of the **business**, and any supporting actor(s), in terms of their **business behaviour**	Describes an **interaction** involving the primary actor, the relevant parts of the **system**, and any supporting actor(s), in terms of their **system behaviour** . In the case of the primary actor, this means only their actions **detectable by the system,** such as making selections, supplying data etc.
How's it executed?	May involve **many organisation units**, systems (or not), technologies, manual / mental procedures etc.	Executed by **automated steps** in the system
Duration	Of **Varying duration** - May be very brief or very long-duration.	Typically quite **short duration** (Cockburn's "coffee-break" rule)

Workflow



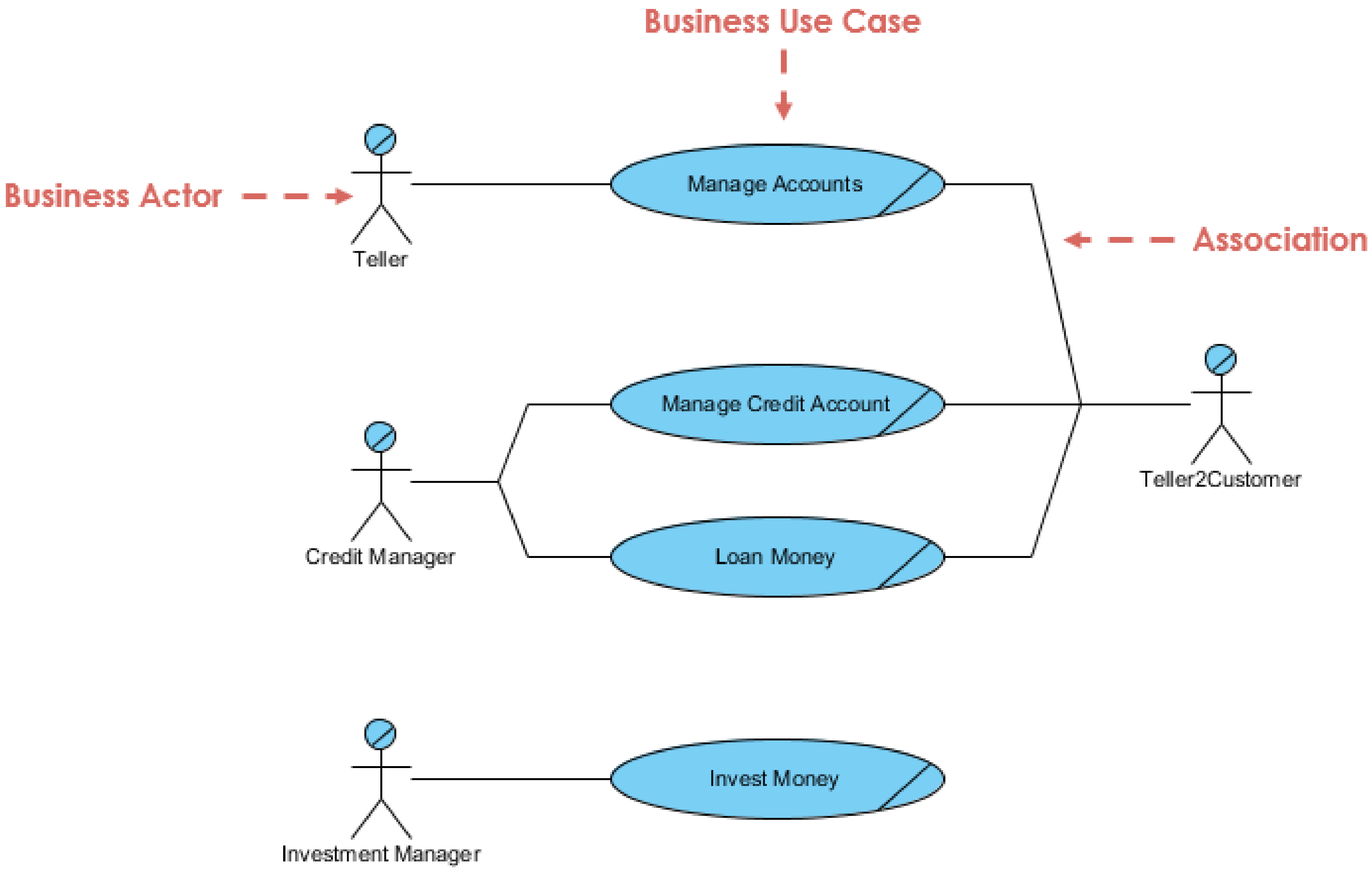
(ICONIX PROCESS)

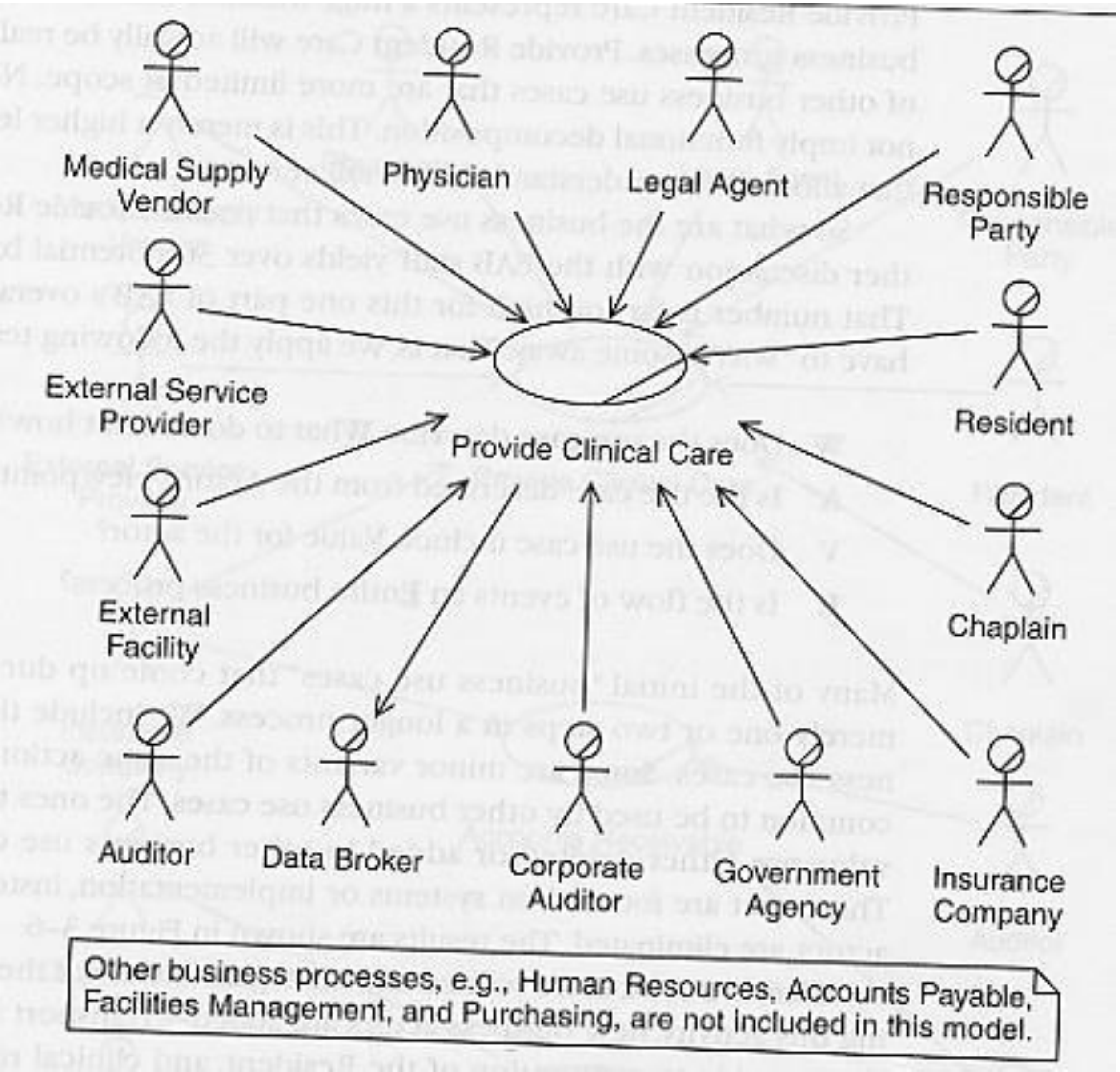
Business Use-Case



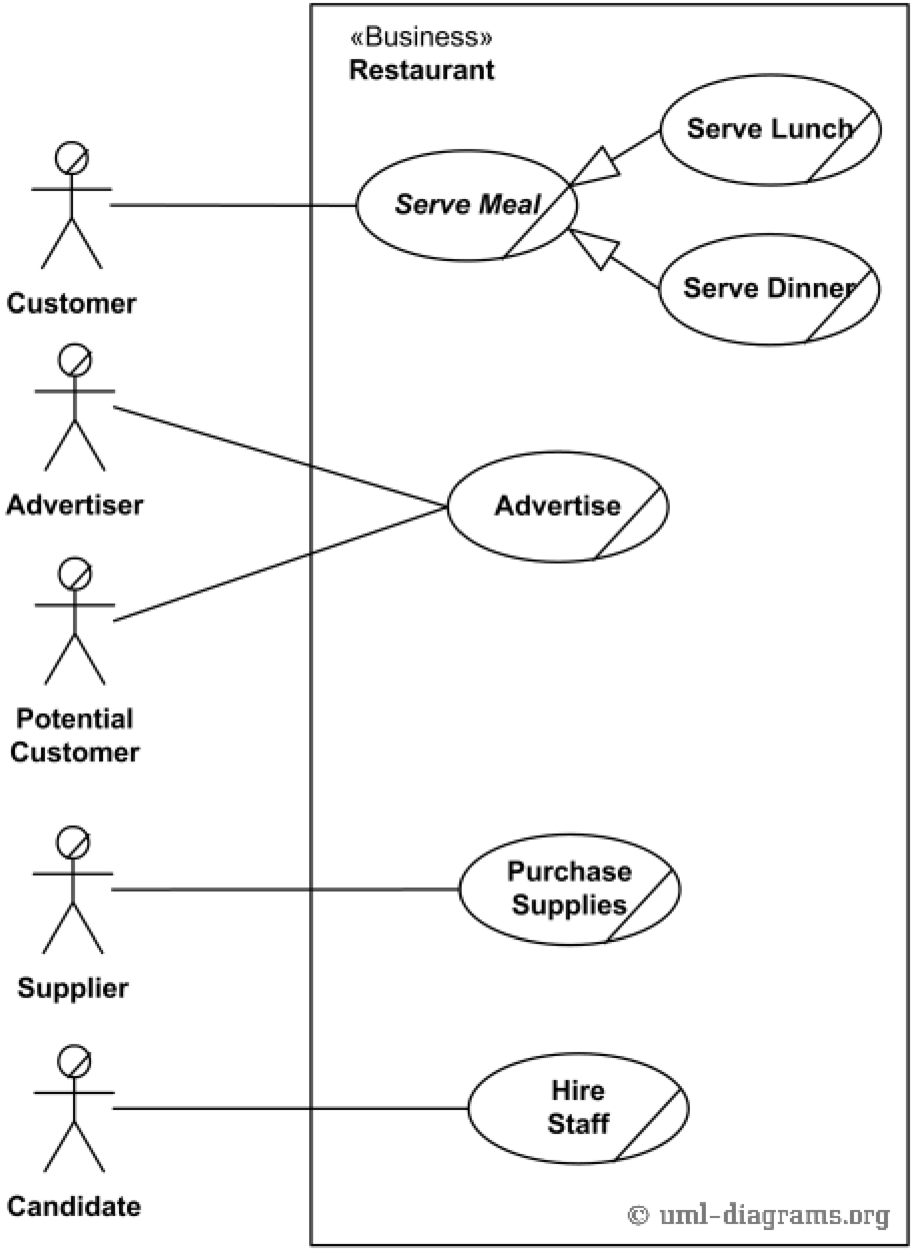
Elements of Business Use Cases

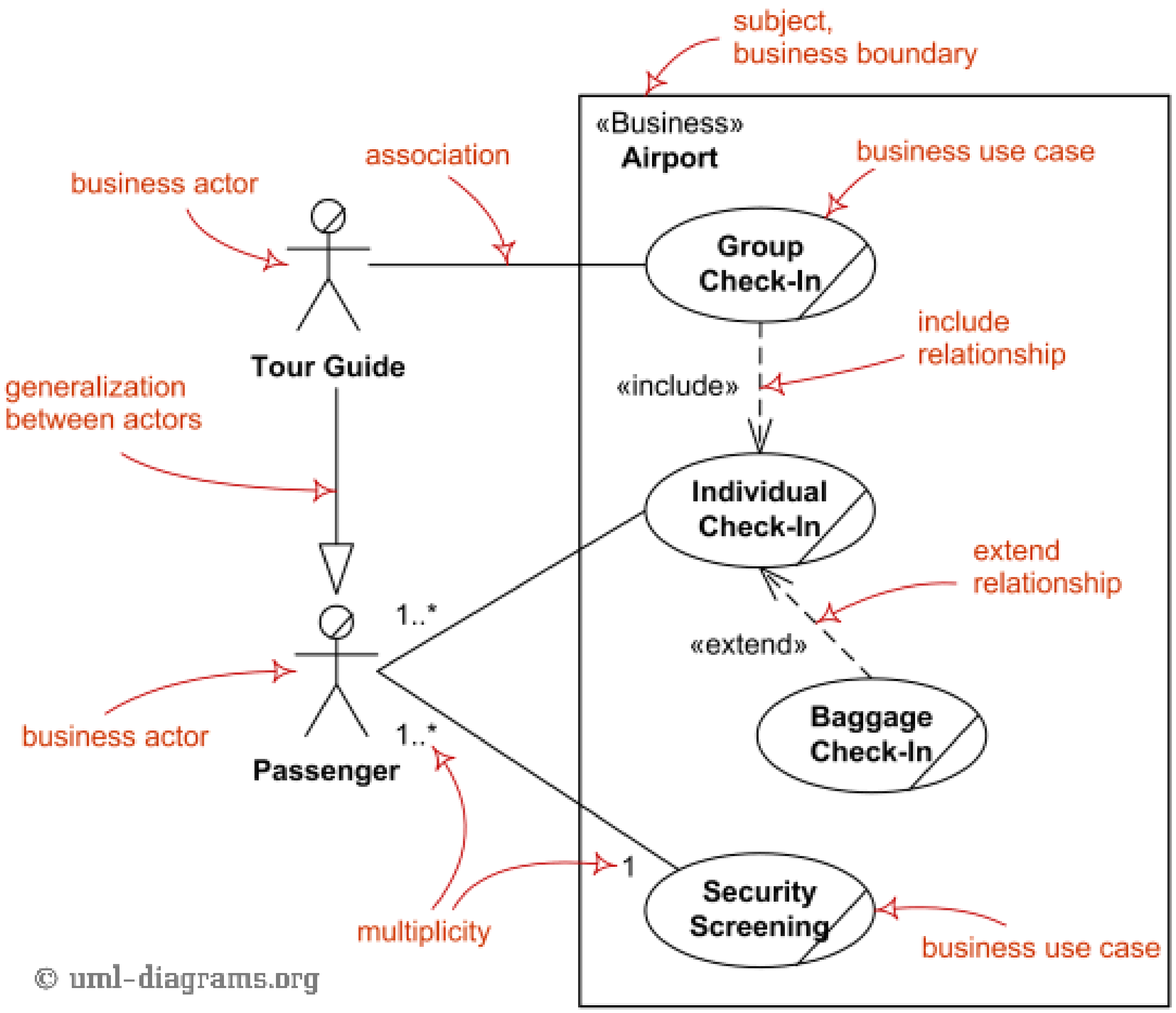
Element	Representation	Notation
Business actors	A business actor is anyone or anything that is *external* to the organization but interacts with it (Individual, group, company,...) *Secondary Actor*	
Business use cases	It represents the workflow within the organization. It keeps focus on what the business is *doing* Named in the form of *<verb><noun>*	
Business workers	A business worker is a role within the organization *Primary Actor*	
Associations	An arrow from a business actor or a business worker to a use case suggests that the actor or worker initiates the use case	

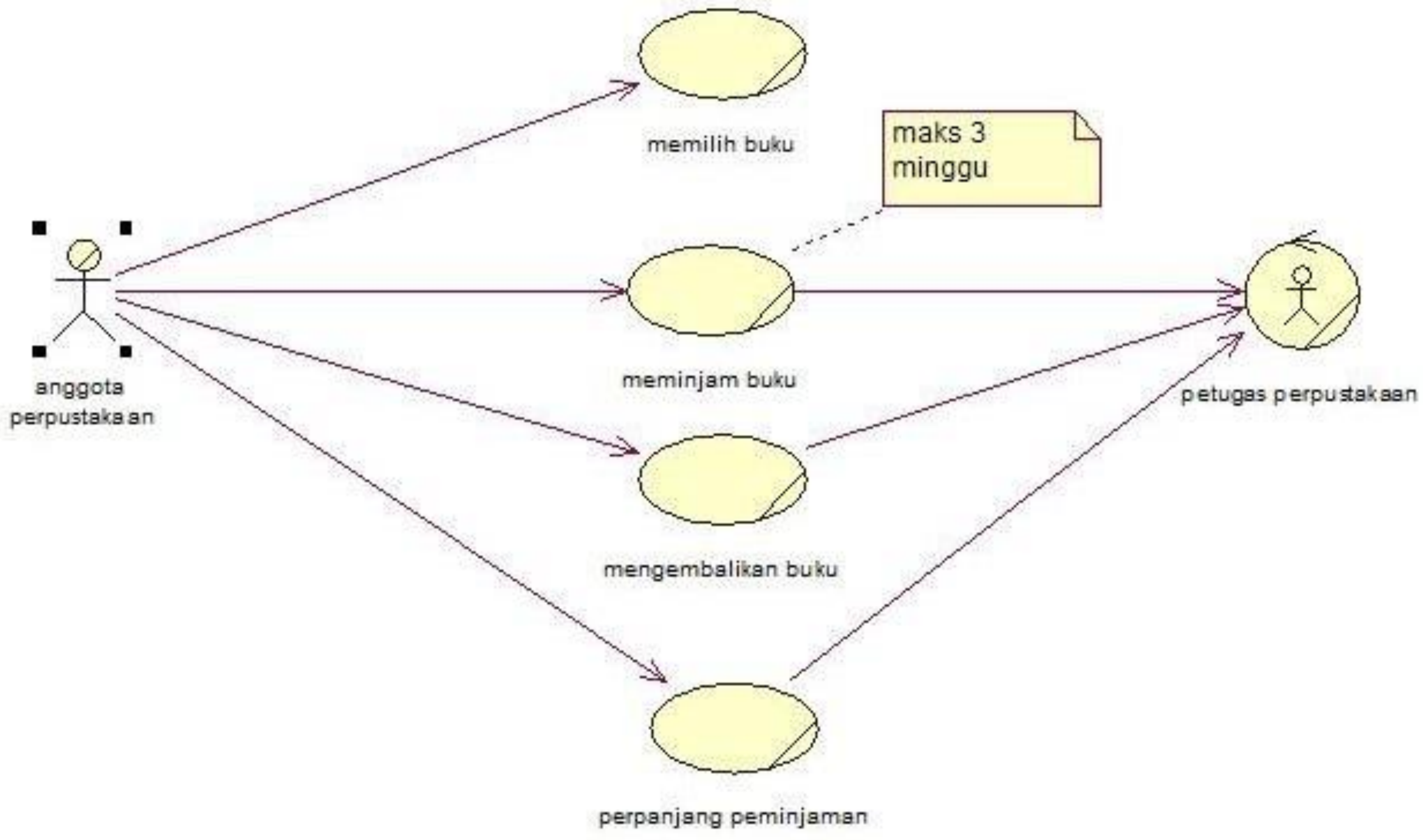

Business Use Case
Business Actor
Association
Teller
Manage Accounts
Manage Credit Account
Credit Manager
Loan Money
Teller2Customer
Investment Manager
Invest Money

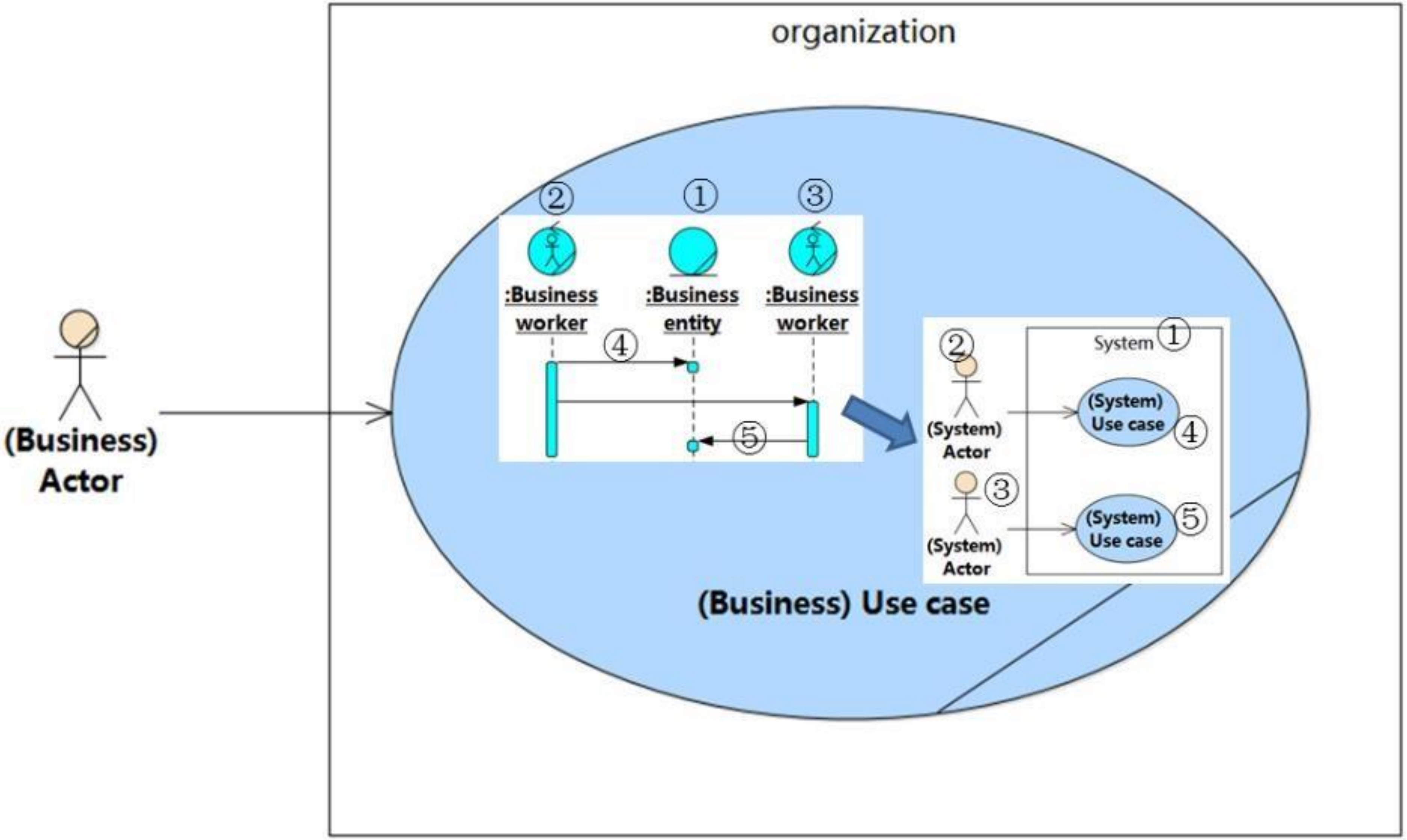


Build up a context of the system to be examined

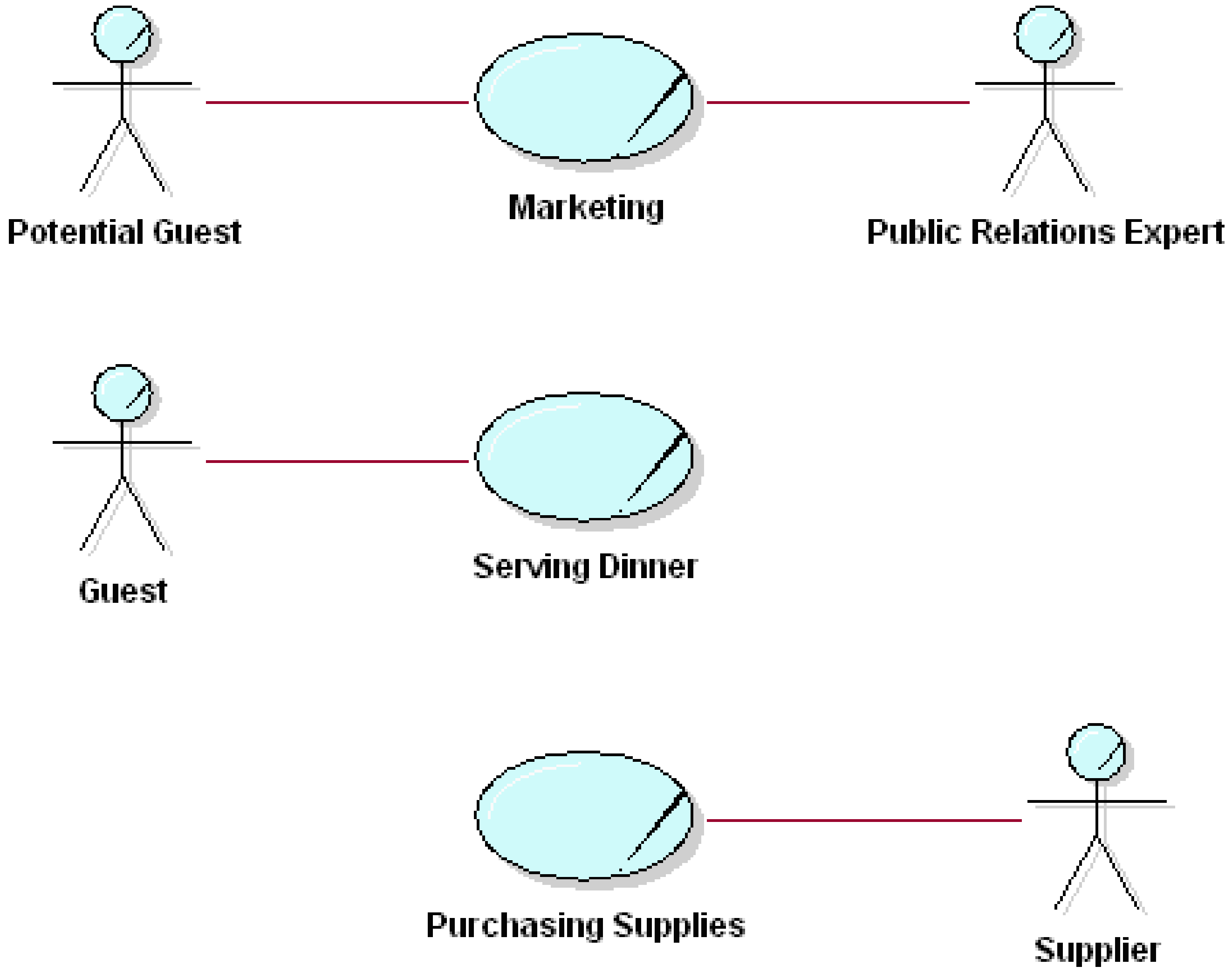

«Business»
Restaurant
Serve Lunch
Serve Meal
Serve Dinner
Customer
Advertiser
Advertise
Potential Customer
Supplier
Purchase Supplies
Candidate
Hire Staff
© uml-diagrams.org


subject,
business boundary
«Business»
Airport
association
business actor
Tour Guide
business use case
Group Check-In
include relationship
«include»
generalization between actors
Individual Check-In
extend relationship
«extend»
1..*
Baggage Check-In
business actor
Passenger
1..*
1
Security Screening
business use case
multiplicity
© uml-diagrams.org

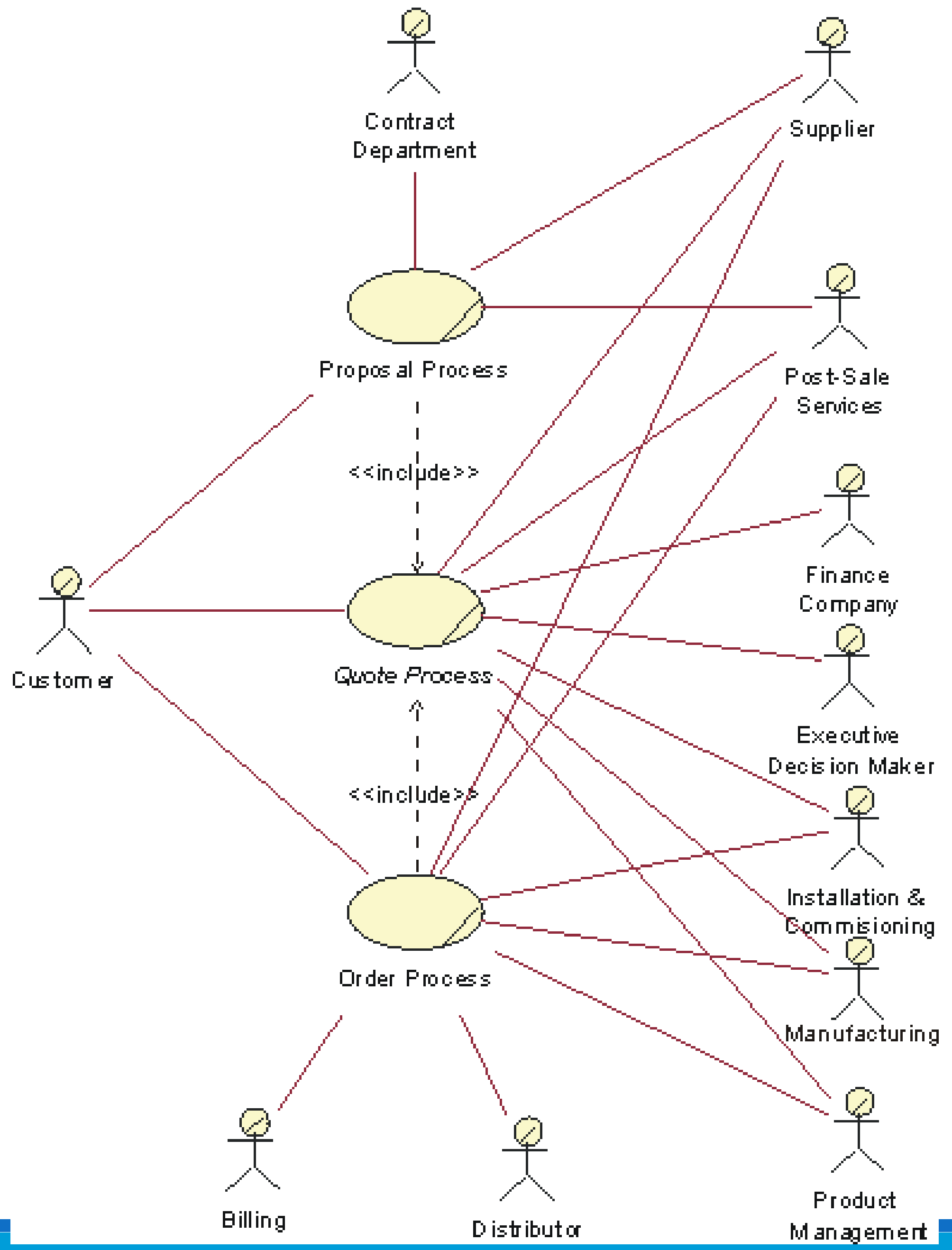

memilih buku
maks 3 minggu
meminjam buku
anggota perpustakaan
petugas perpustakaan
mengembalikan buku
perpanjang peminjaman



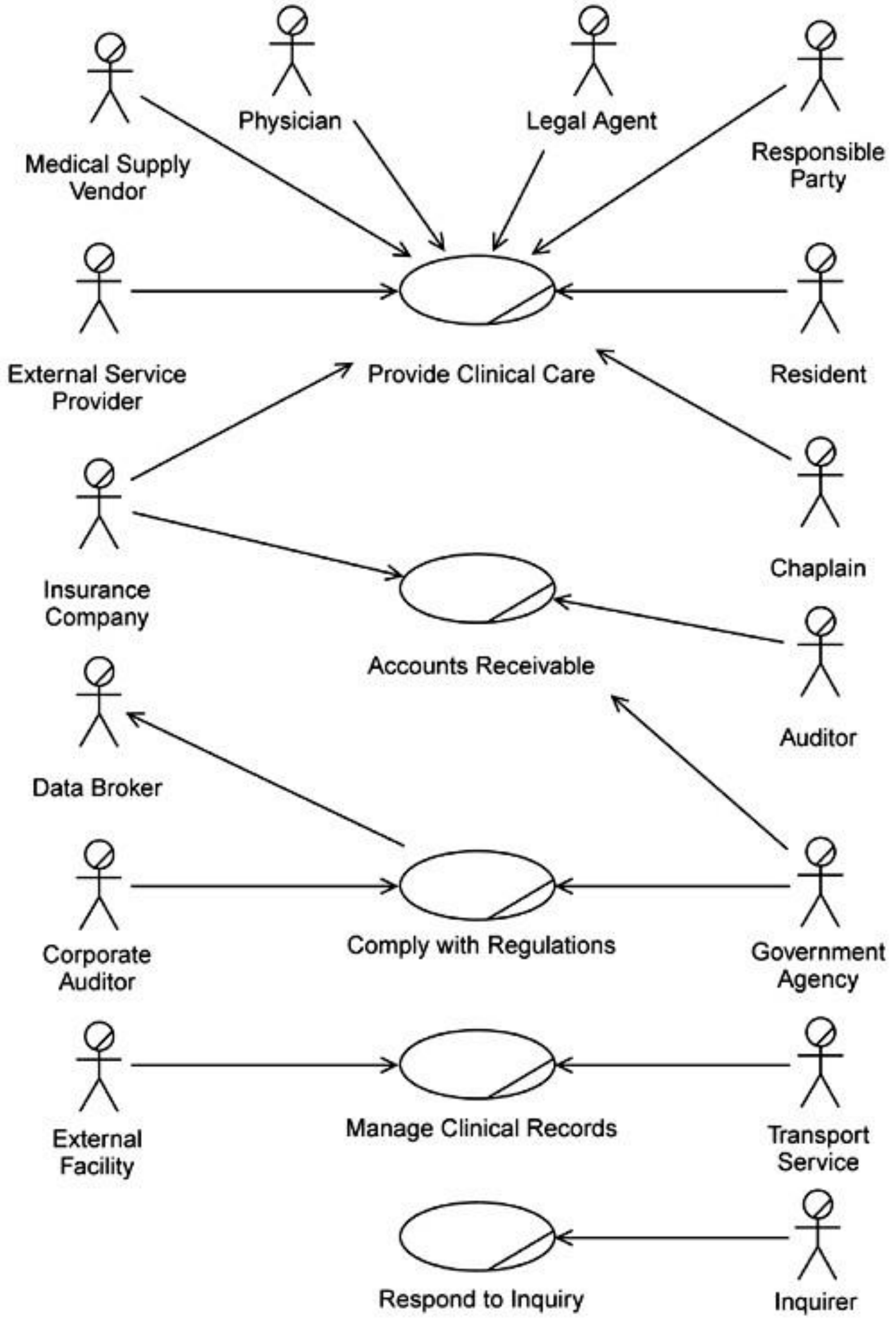
BUSINESS USE-CASE & USE-CASE DIAGRAM

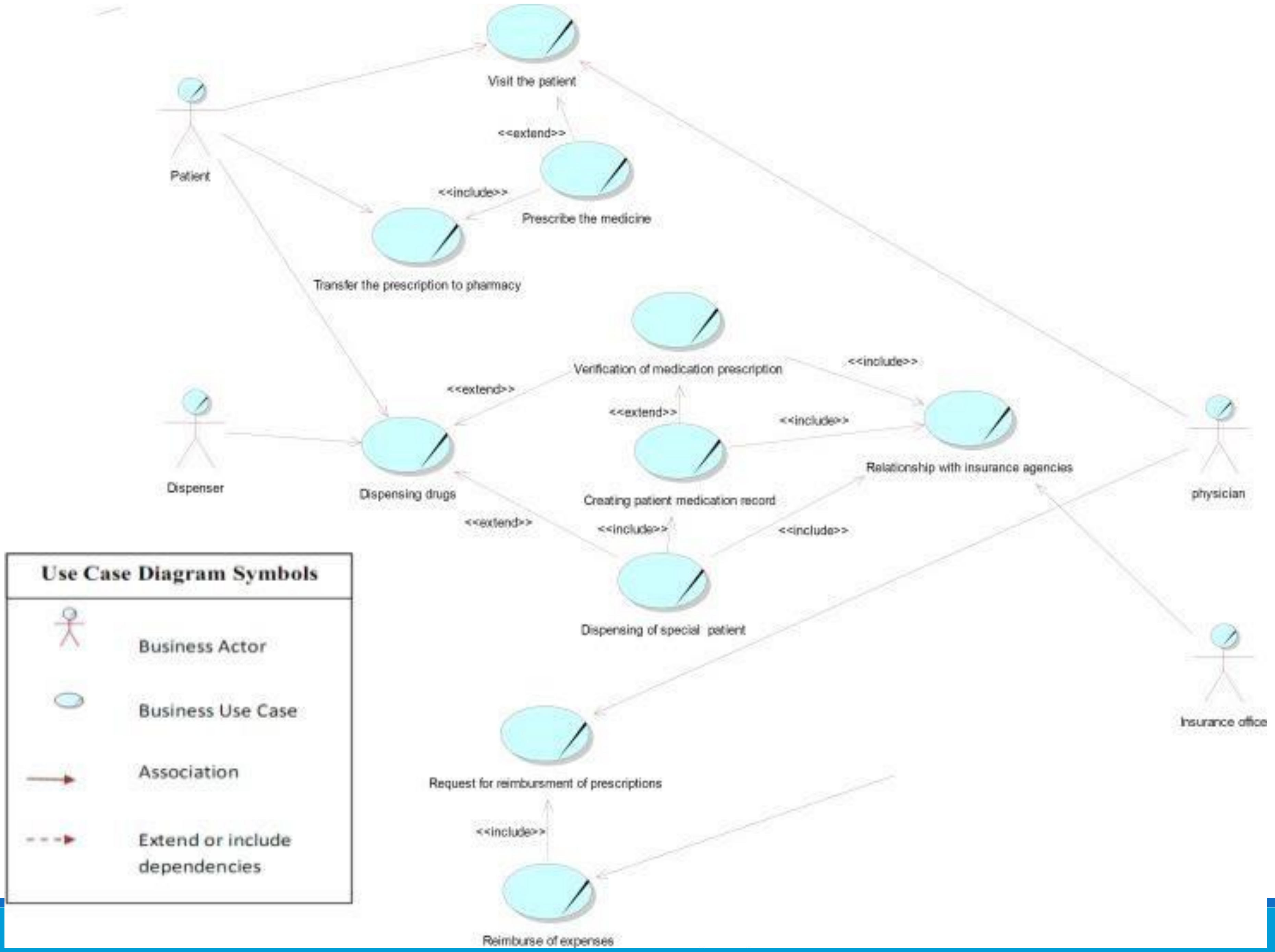


BUSINESS ACTOR



BUSINESS ACTOR


Medical Supply Vendor
Physician
Legal Agent
Responsible Party
External Service Provider
Provide Clinical Care
Resident
Insurance Company
Chaplain
Accounts Receivable
Auditor
Data Broker
Corporate Auditor
Comply with Regulations
Government Agency
External Facility
Manage Clinical Records
Transport Service
Respond to Inquiry
Inquirer


Visit the patient
<<extend>>
Patient
<<include>>
Prescribe the medicine
Transfer the prescription to pharmacy
Verification of medication prescription
<<include>>
<<extend>>
<<extend>>
<<include>>
Dispenser
Dispensing drugs
Creating patient medication record
Relationship with insurance agencies
physician
<<extend>>
<<include>>
<<include>>
Dispensing of special patient
Insurance office
Request for reimbursment of prescriptions
<<include>>
Reimburse of expenses
Use Case Diagram Symbols
Business Actor
Business Use Case
Association
Extend or include dependencies

Use-Case Diagram

Requirement Analysis

1. Identify the business processes.
2. Identify the IT processes that support each of the business processes.
3. Identify the activities within each of the IT processes.
4. Identify the functions within each of the activities.
5. Identify the use cases for one or more of the functions.

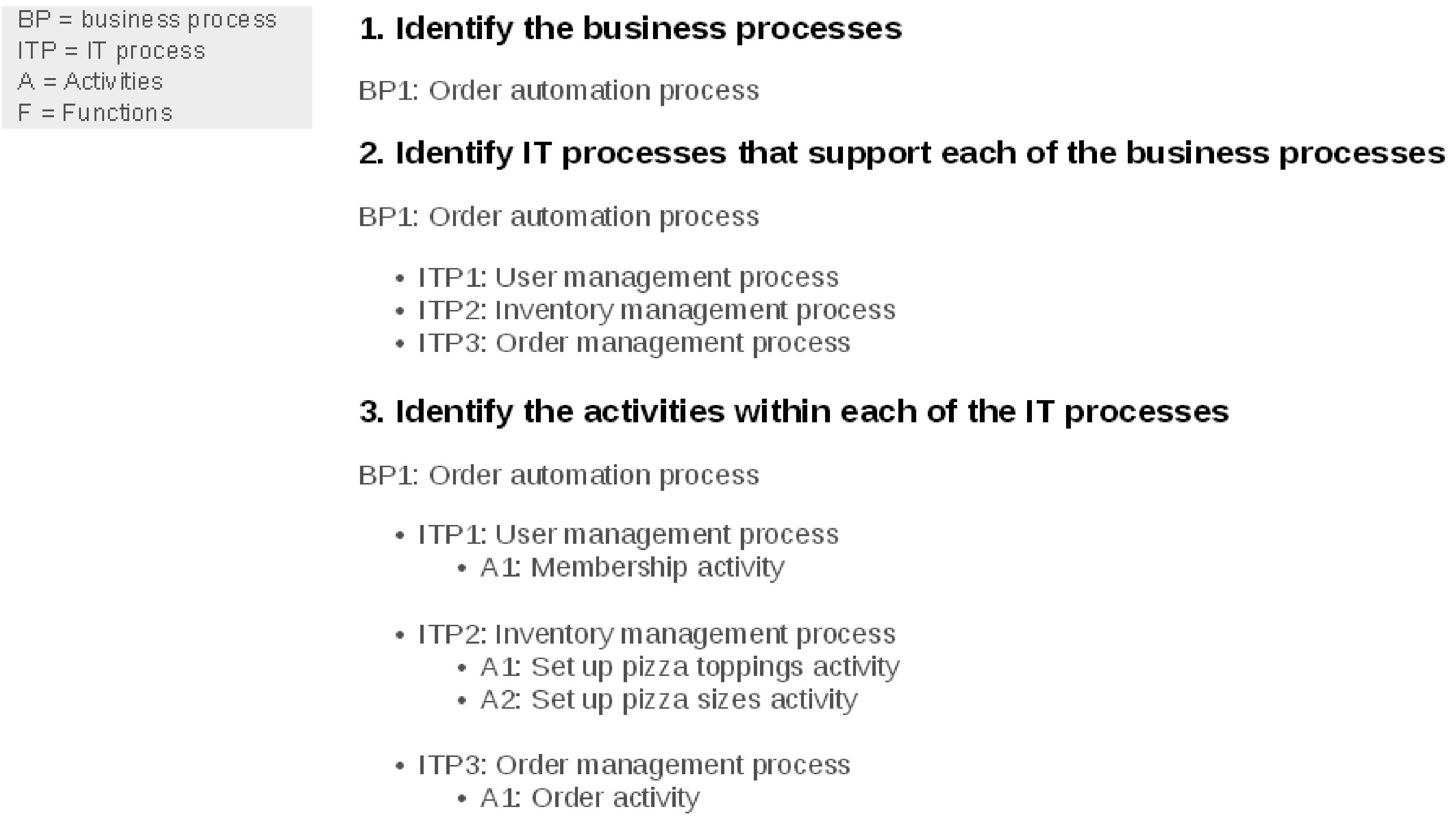
Example: Online pizza ordering system

A corner gourmet pizza vendor, who has operated a traditional pizza delivery service using telephone orders, wants to automate the ordering process by developing an online system.

The customers need to be able to:

- Select the pizza toppings, size, and number of pizzas.
- Log in and enter the delivery address.
- Specify the time of delivery.
- Revise or delete their orders.

Requirement Analysis



BP = business process
ITP = IT process
A = Activities
F = Functions

1. Identify the business processes

BP1: Order automation process

2. Identify IT processes that support each of the business processes

BP1: Order automation process

- ITP1: User management process
- ITP2: Inventory management process
- ITP3: Order management process

3. Identify the activities within each of the IT processes

BP1: Order automation process

- ITP1: User management process
    - A1: Membership activity
- ITP2: Inventory management process
    - A1: Set up pizza toppings activity
    - A2: Set up pizza sizes activity
- ITP3: Order management process
    - A1: Order activity

Business Use-Case Diagram

4. Identify the functions within each of the activities

BP1: Order automation process

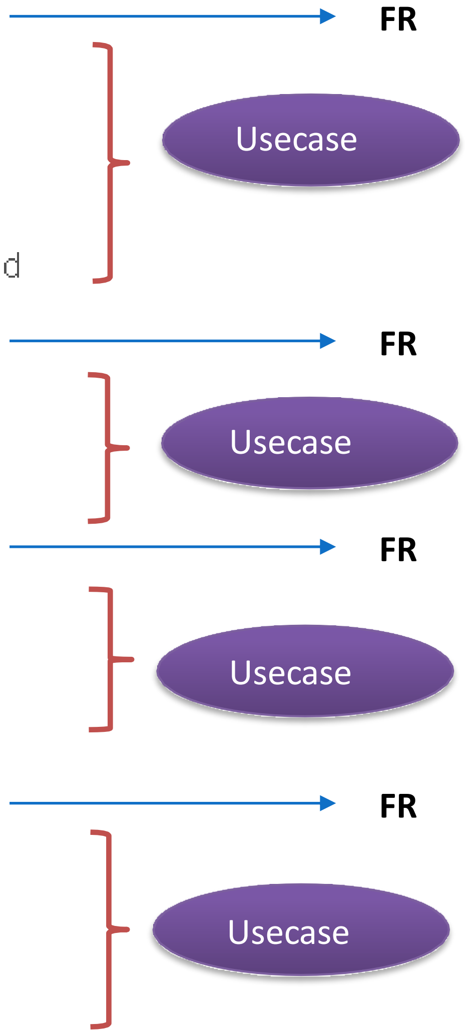
- ITP1: User management process
  - A1: Membership activity

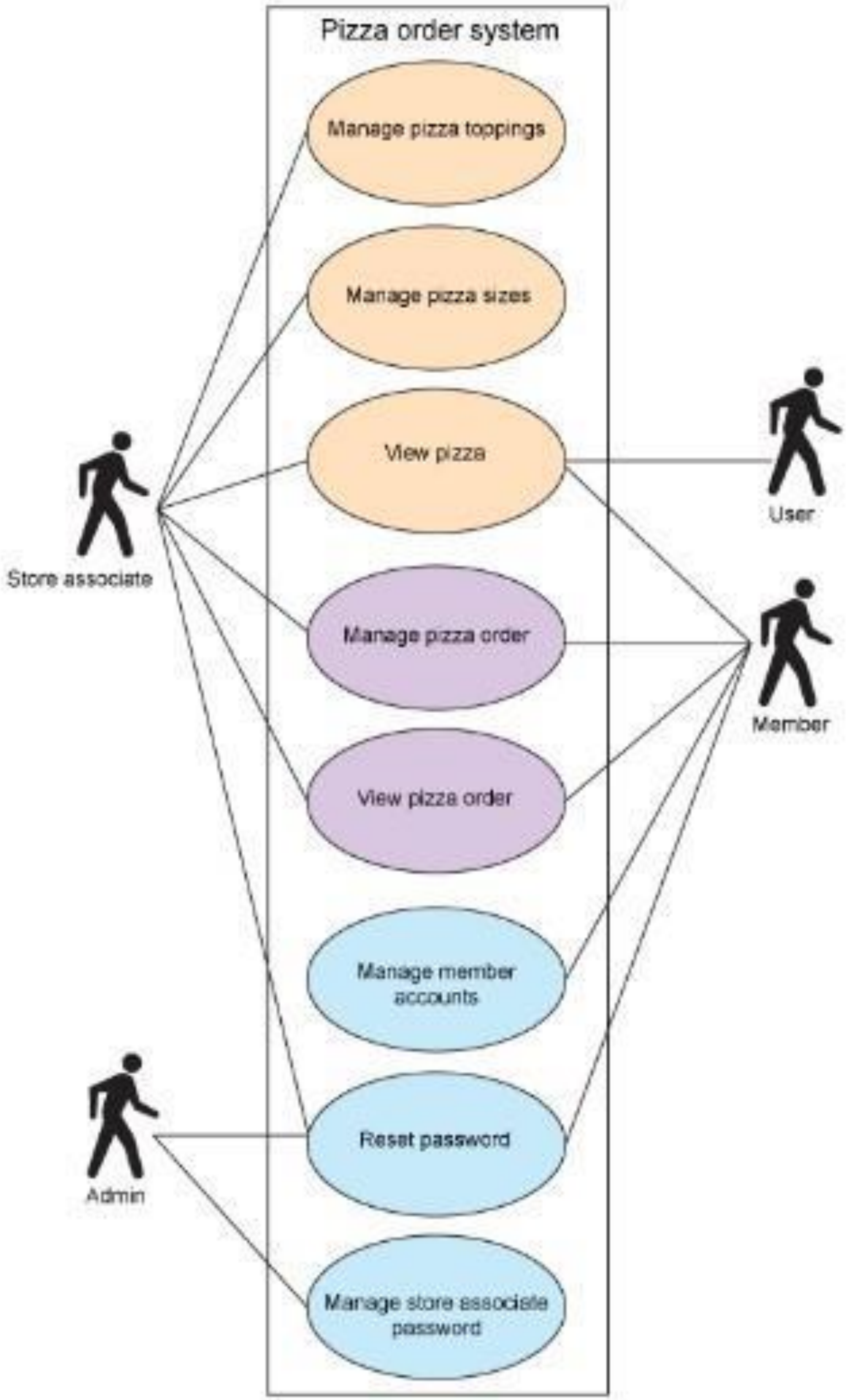
    F1: Create member
    F2: Update member
    F3: Delete member
    F4: View members
    F5: Reset password
    F6: Create store associate password
- ITP2: Inventory management process
  - A1: Set up pizza toppings activity

    F1: Add pizza topping and price
    F2: Update pizza topping
    F3: Delete pizza topping
    F4: View pizza toppings
  - A2: Set up pizza sizes activity

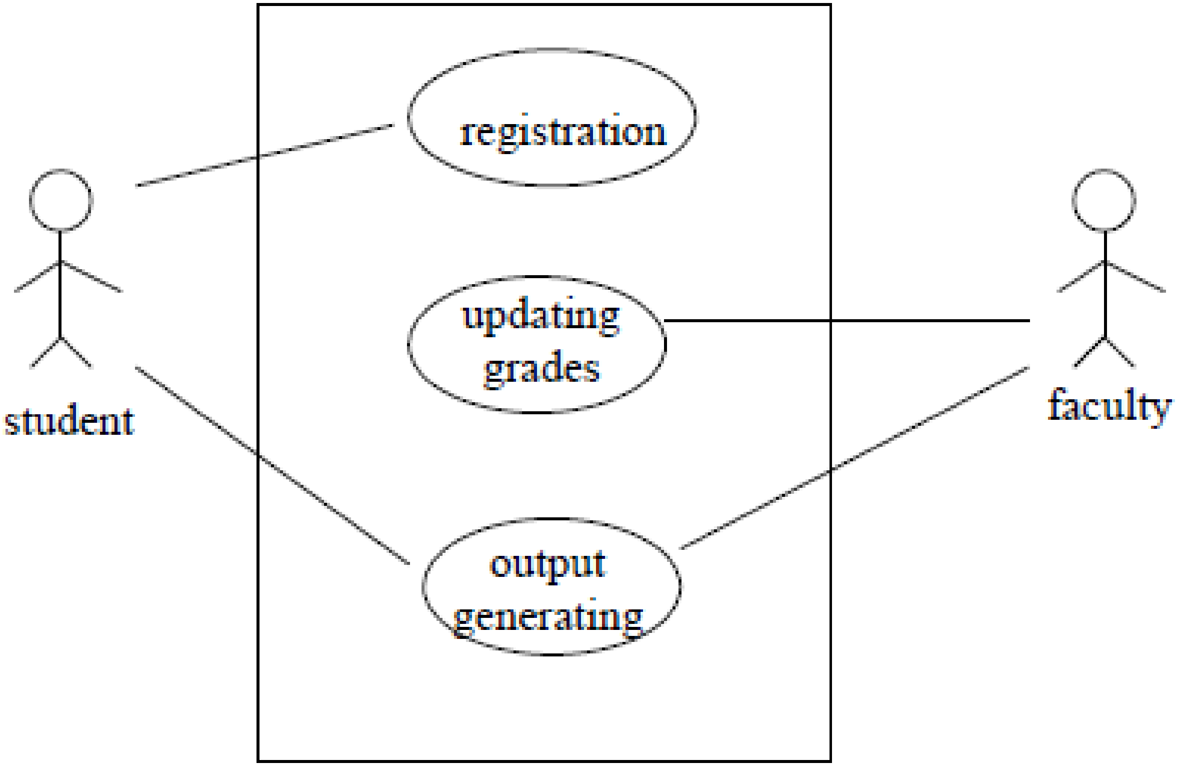
    F1: Add pizza size and price
    F2: Update pizza size
    F3: Delete pizza size
    F4: View pizza sizes
- ITP3: Order management process
  - A1: Order activity

    F1: Enter order
    F2: View order
    F3: Submit order
    F4: Revise order
    F5: Delete order

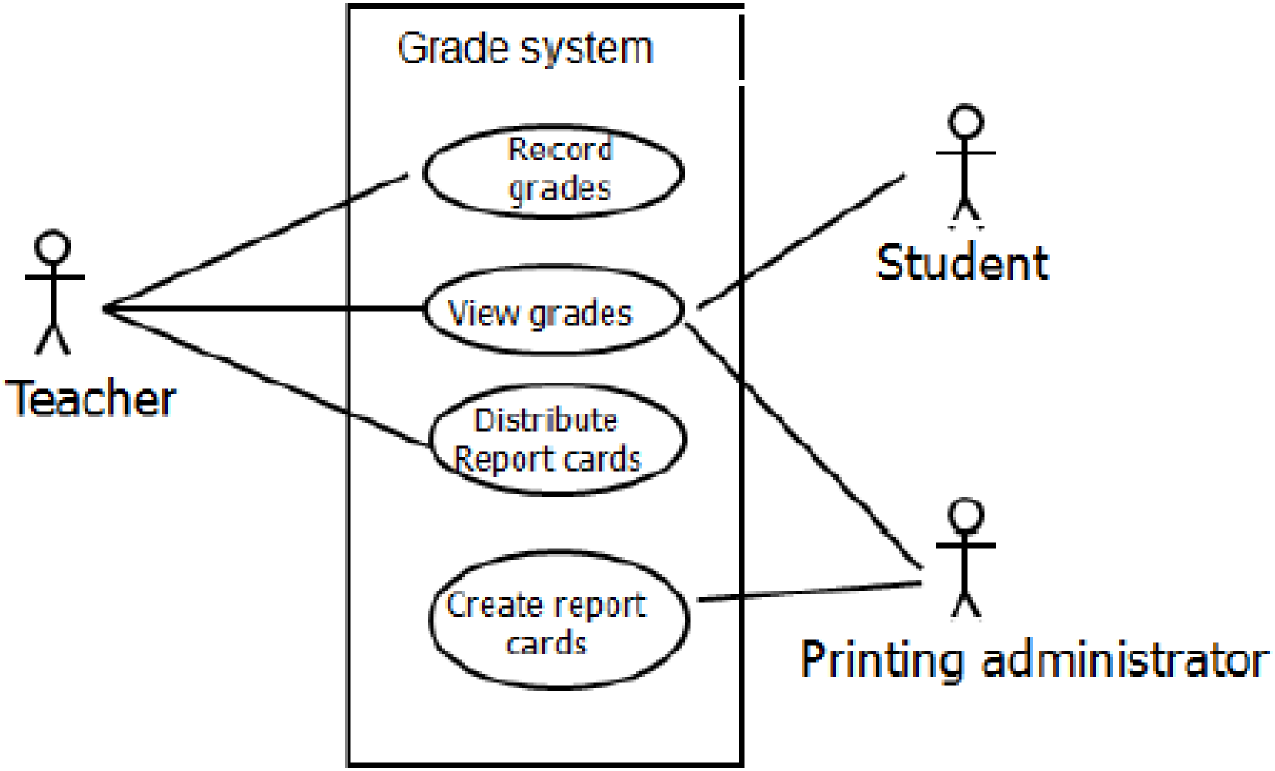


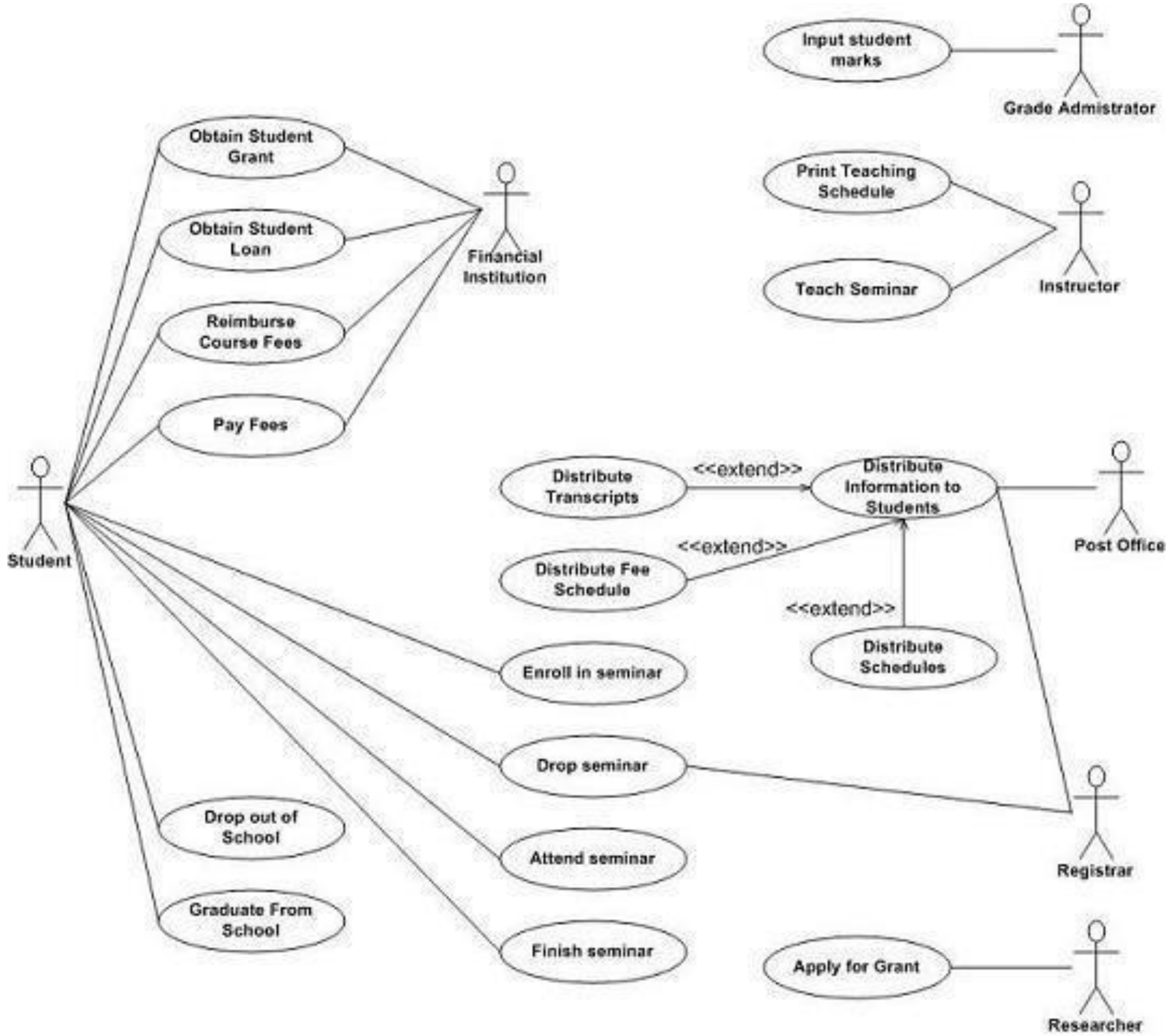

Pizza order system
Manage pizza toppings
Manage pizza sizes
View pizza
Manage pizza order
View pizza order
Manage member accounts
Reset password
Manage store associate password
Store associate
User
Member
Admin

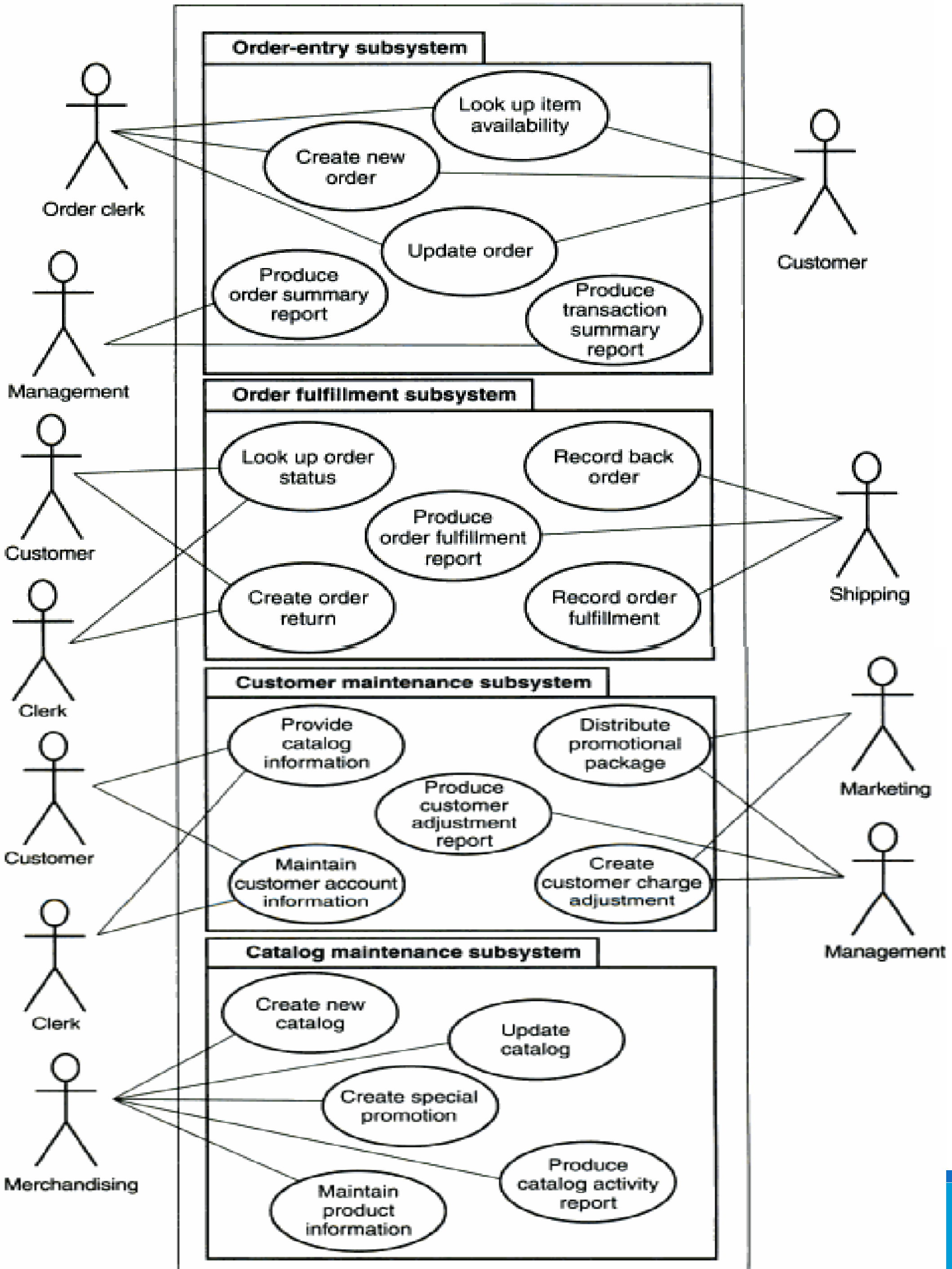
Use-Case Diagram Example



Use-Case Diagram Example




Input student marks
Grade Admistrator
Obtain Student Grant
Obtain Student Loan
Reimburse Course Fees
Pay Fees
Financial Institution
Print Teaching Schedule
Teach Seminar
Instructor
Student
Distribute Transcripts
<<extend>>
Distribute Information to Students
Post Office
Distribute Fee Schedule
<<extend>>
<<extend>>
Distribute Schedules
Enroll in seminar
Drop seminar
Registrar
Drop out of School
Attend seminar
Graduate From School
Finish seminar
Apply for Grant
Researcher

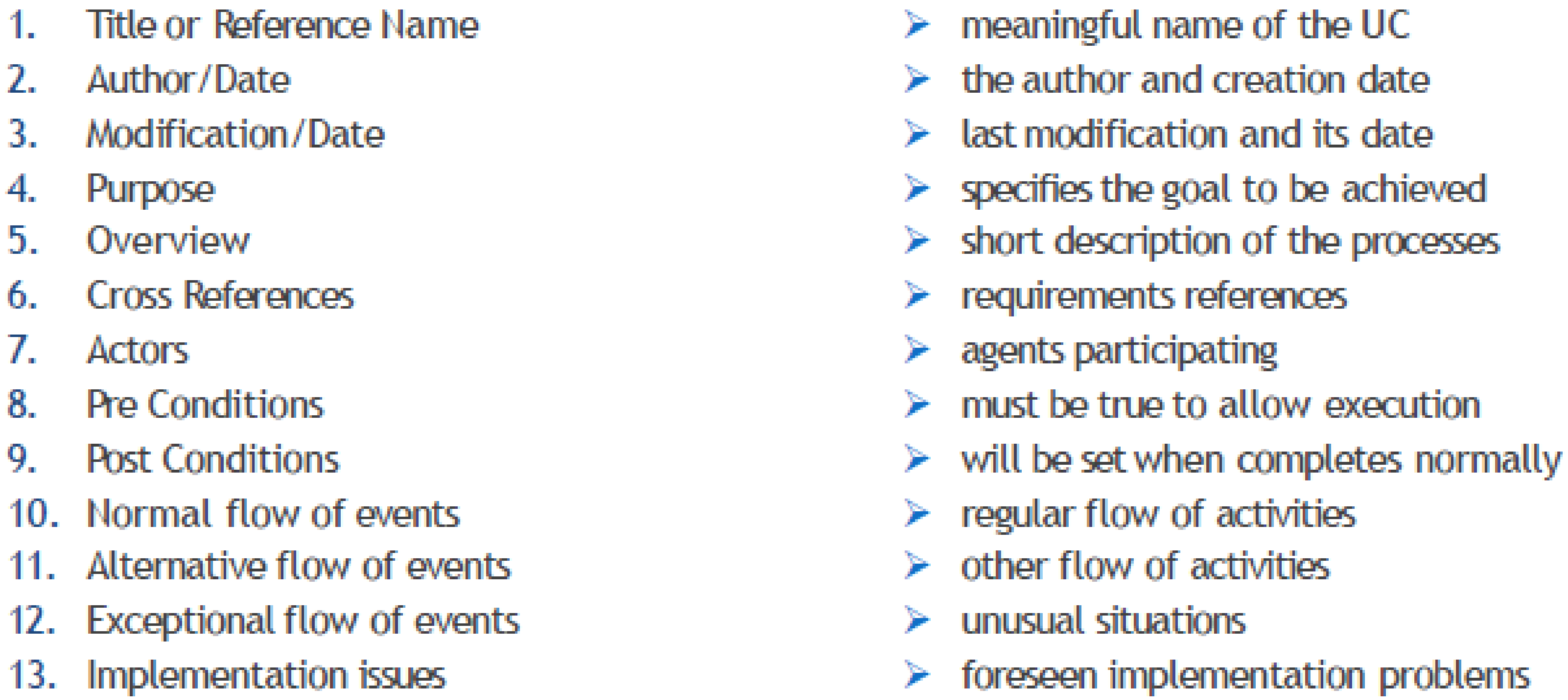

Order-entry subsystem
Look up item availability
Create new order
Update order
Produce order summary report
Produce transaction summary report
Order clerk
Management
Customer
Order fulfillment subsystem
Look up order status
Record back order
Produce order fulfillment report
Create order return
Record order fulfillment
Customer
Clerk
Shipping
Customer maintenance subsystem
Provide catalog information
Distribute promotional package
Produce customer adjustment report
Maintain customer account information
Create customer charge adjustment
Customer
Clerk
Marketing
Management
Catalog maintenance subsystem
Create new catalog
Update catalog
Create special promotion
Produce catalog activity report
Maintain product information
Merchandising

Use-Case Specification

Definition

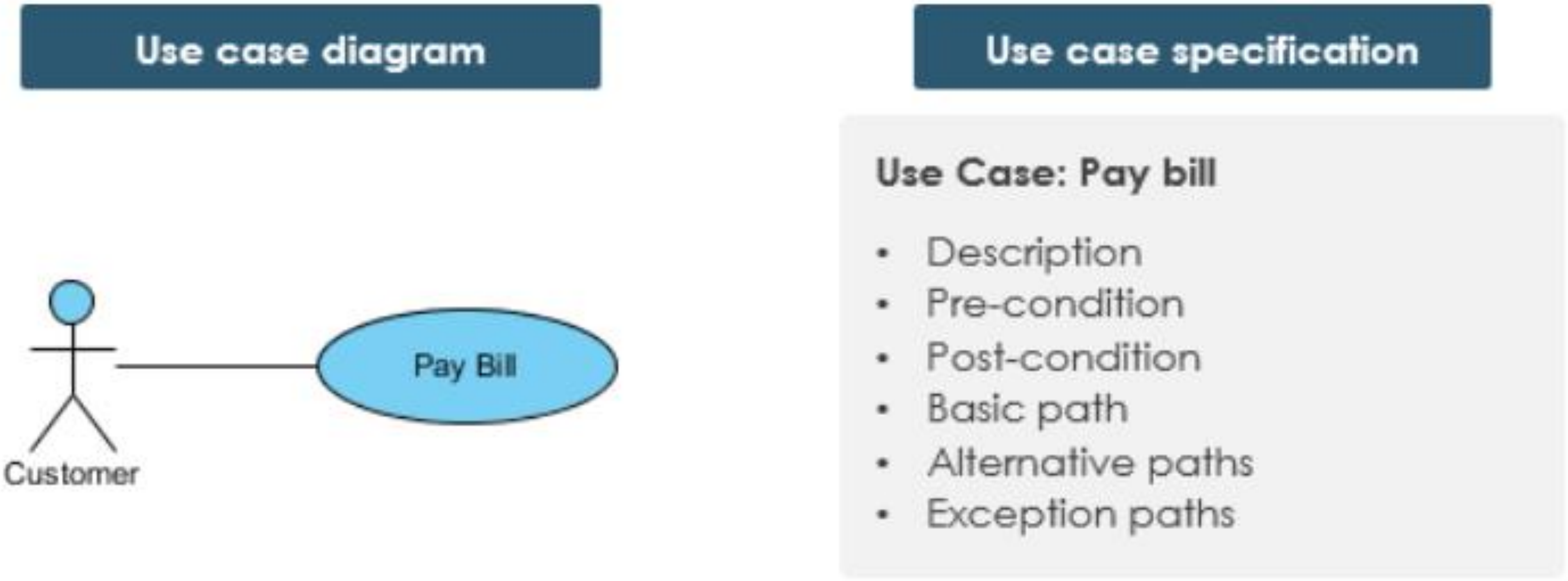
A use case specification **captures the requirements**, typically of a system, in the form of a use case that **contains the descriptive requirements steps in a logical sequence**so that document specification can be understood by users to obtain sign-off of their requirements and for testers and developers to understand what is needed by the system to test and build the system functionality detailed in the system use usecase.

Each use case may include..



1. Title or Reference Name	➢ meaningful name of the UC
2. Author/Date	➢ the author and creation date
3. Modification/Date	➢ last modification and its date
4. Purpose	➢ specifies the goal to be achieved
5. Overview	➢ short description of the processes
6. Cross References	➢ requirements references
7. Actors	➢ agents participating
8. Pre Conditions	➢ must be true to allow execution
9. Post Conditions	➢ will be set when completes normally
10. Normal flow of events	➢ regular flow of activities
11. Alternative flow of events	➢ other flow of activities
12. Exceptional flow of events	➢ unusual situations
13. Implementation issues	➢ foreseen implementation problems

Use-Case Specification

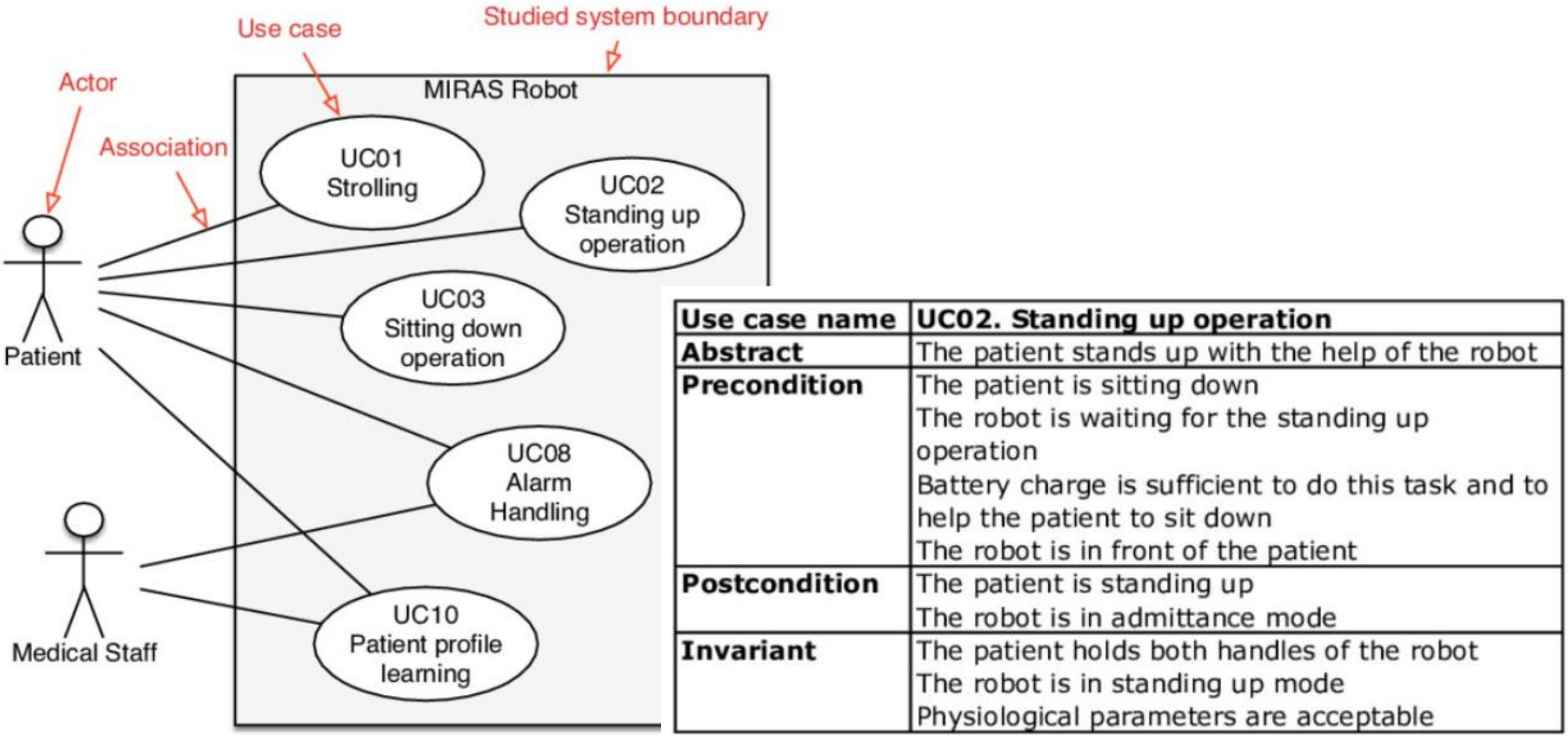


Use case diagram

Use case specification

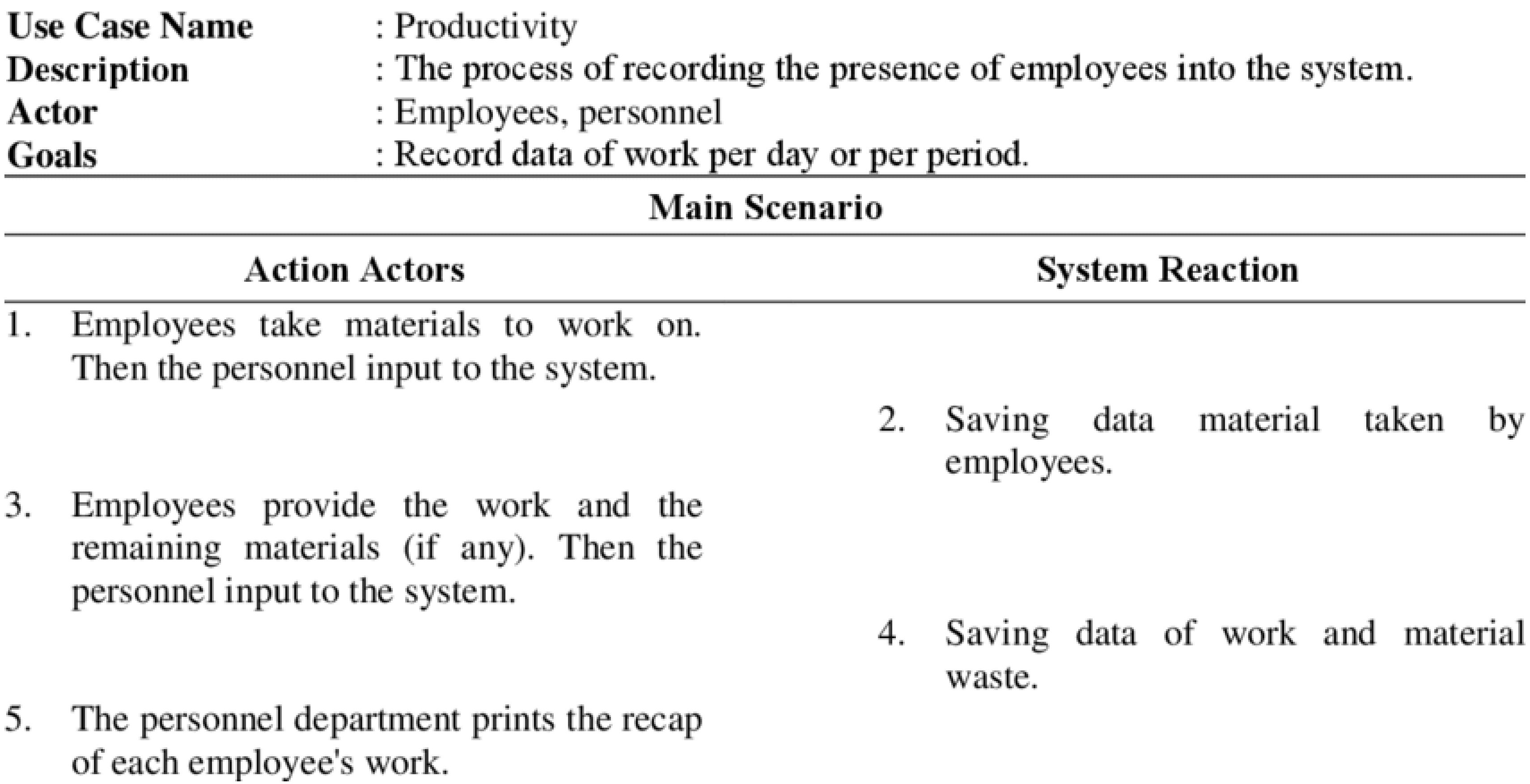
Use Case: Pay bill

- Description
- Pre-condition
- Post-condition
- Basic path
- Alternative paths
- Exception paths



Use case name	UC02. Standing up operation
Abstract	The patient stands up with the help of the robot
Precondition	The patient is sitting down The robot is waiting for the standing up operation Battery charge is sufficient to do this task and to help the patient to sit down The robot is in front of the patient
Postcondition	The patient is standing up The robot is in admittance mode
Invariant	The patient holds both handles of the robot The robot is in standing up mode Physiological parameters are acceptable

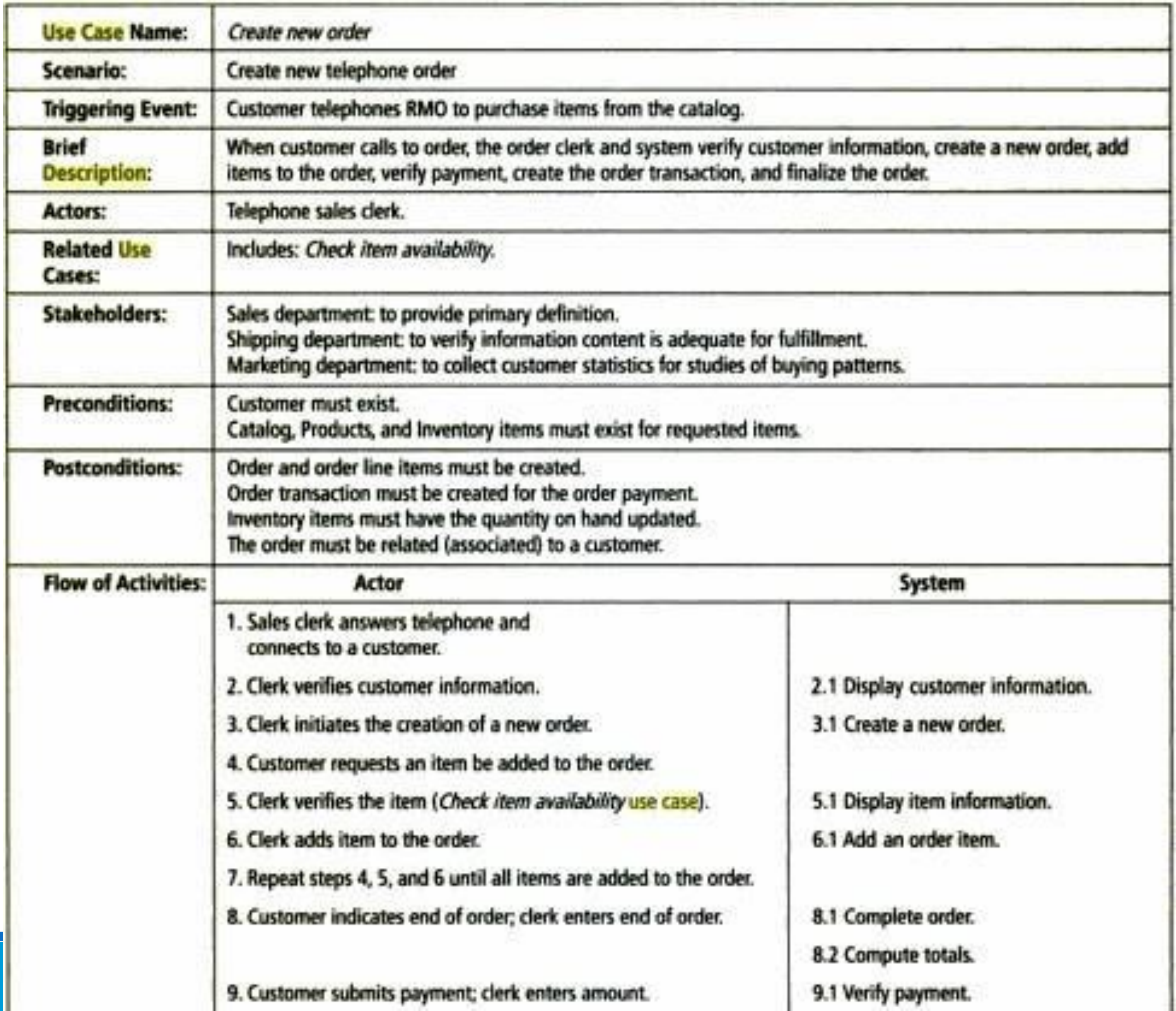
Use-Case Scenario



Use Case Name : Productivity
Description : The process of recording the presence of employees into the system.
Actor : Employees, personnel
Goals : Record data of work per day or per period.

Main Scenario

Action Actors	System Reaction
1. Employees take materials to work on. Then the personnel input to the system.	
	2. Saving data material taken by employees.
3. Employees provide the work and the remaining materials (if any). Then the personnel input to the system.	
	4. Saving data of work and material waste.
5. The personnel department prints the recap of each employee's work.	



Use Case Name:	*Create new order*	
Scenario:	Create new telephone order	
Triggering Event:	Customer telephones RMO to purchase items from the catalog.	
Brief Description:	When customer calls to order, the order clerk and system verify customer information, create a new order, add items to the order, verify payment, create the order transaction, and finalize the order.	
Actors:	Telephone sales clerk.	
Related Use Cases:	Includes: *Check item availability.*	
Stakeholders:	Sales department: to provide primary definition. Shipping department: to verify information content is adequate for fulfillment. Marketing department: to collect customer statistics for studies of buying patterns.	
Preconditions:	Customer must exist. Catalog, Products, and Inventory items must exist for requested items.	
Postconditions:	Order and order line items must be created. Order transaction must be created for the order payment. Inventory items must have the quantity on hand updated. The order must be related (associated) to a customer.	
Flow of Activities:	**Actor**	**System**
	1. Sales clerk answers telephone and connects to a customer.	
	2. Clerk verifies customer information.	2.1 Display customer information.
	3. Clerk initiates the creation of a new order.	3.1 Create a new order.
	4. Customer requests an item be added to the order.	
	5. Clerk verifies the item (*Check item availability* use case).	5.1 Display item information.
	6. Clerk adds item to the order.	6.1 Add an order item.
	7. Repeat steps 4, 5, and 6 until all items are added to the order.	
	8. Customer indicates end of order; clerk enters end of order.	8.1 Complete order. 8.2 Compute totals.
	9. Customer submits payment; clerk enters amount.	9.1 Verify payment.

Use-Case Modeling

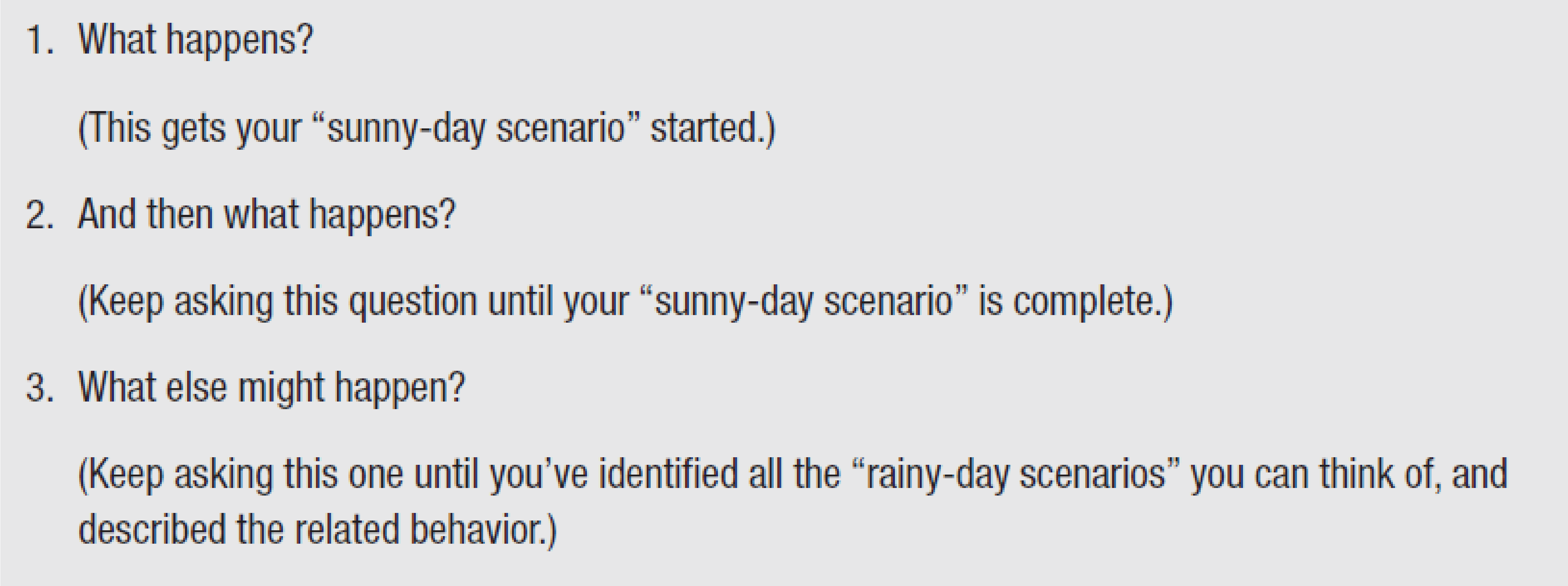
Guidelines

1. Follow the two-paragraph rule.
2. Organize your use cases with actors and use case diagrams.
3. Write your use cases in active voice.
4. Write your use case using an event/response flow, describing both sides of the user/ system dialogue.

1) Follow the two-paragraph rule

- Each use case should fit comfortably into two paragraphs, including both the basic course and alternate courses.
- Anything much longer than two paragraphs will result in some incomprehensible sequence diagrams.
- If your use case goes over two paragraphs, it probably needs to be divided into two or more separate use cases

Fundamental Questions



1. What happens?

   (This gets your “sunny-day scenario” started.)

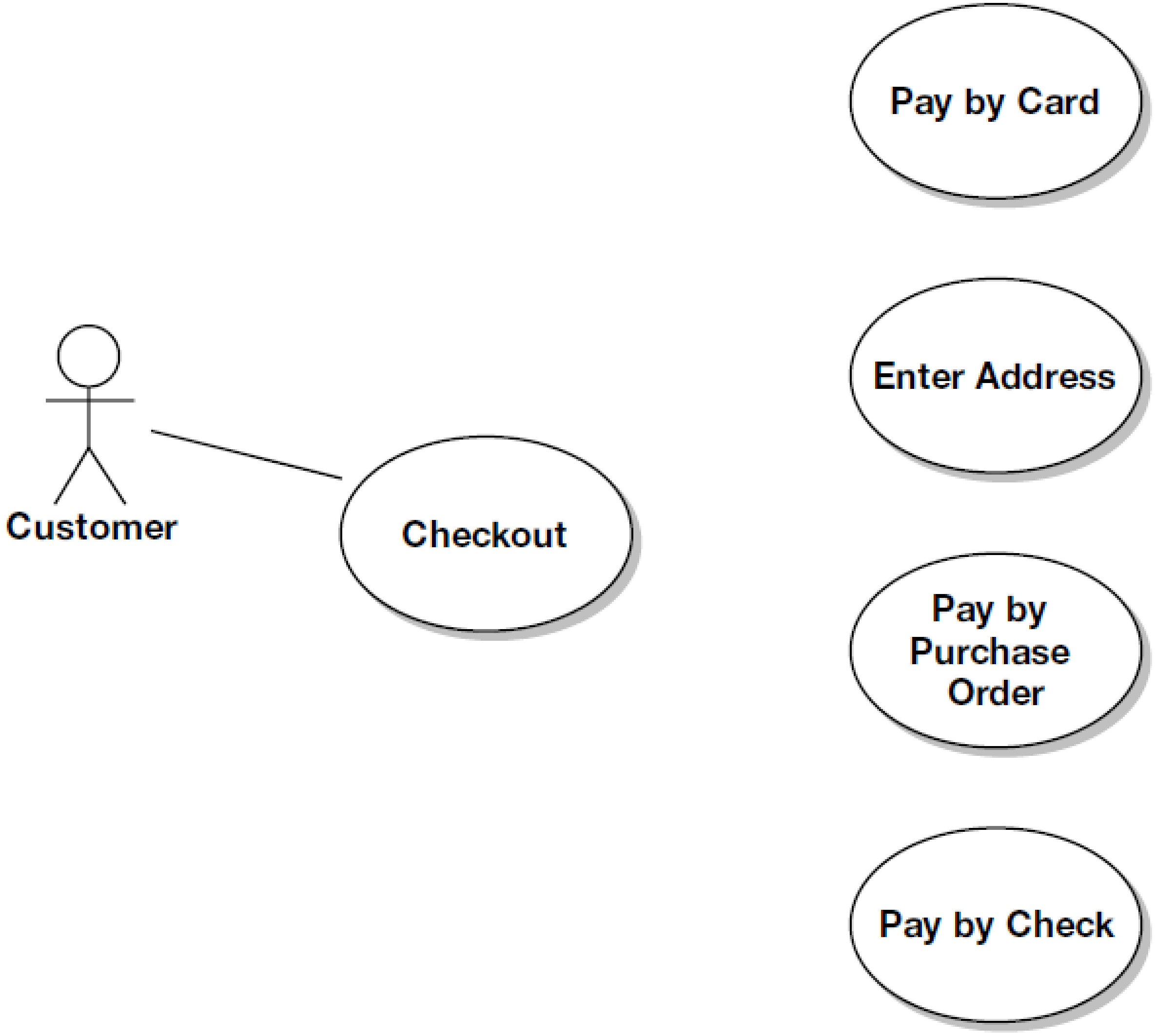
2. And then what happens?

   (Keep asking this question until your “sunny-day scenario” is complete.)

3. What else might happen?

   (Keep asking this one until you’ve identified all the “rainy-day scenarios” you can think of, and described the related behavior.)

2) Organize your use cases with actors and use case diagrams

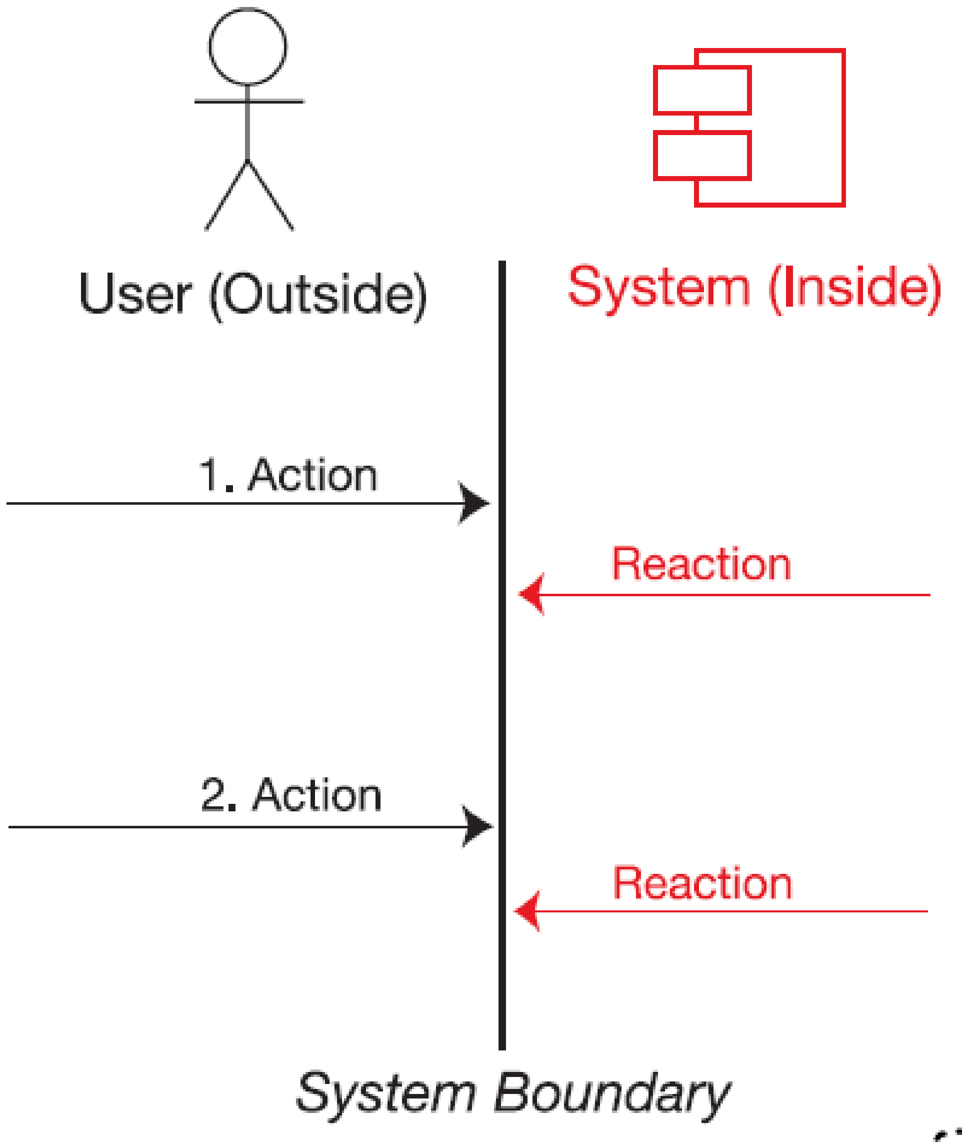


3) Write your use cases in active voice.

1. The capability is provided for users to log in, using a password-protected authorization scheme.

2. The user **enters her username and password**, and then **clicks the Login button**. The system **looks up the user profile** using the username and **checks the password**. The system then **logs in the user**.

4) Write your use case using an event/response flow, describing both sides of the user/ system dialogue



2. The user enters her username and password, and then clicks the Login button. The system looks up the user profile using the username and checks the password. The system then logs in the user.

3. The system displays the Login screen. The user enters her username and password, and then clicks the Login button. The system looks up the user profile using the username and checks the password. The system then logs in the user.

TERIMA KASIH